

ARTIFICIAL INTELLIGENCE RESEARCH AND DEVELOPMENT

VISIT...

LANZAROTE
Caliente.COM

Frontiers in Artificial Intelligence and Applications

FAIA covers all aspects of theoretical and applied artificial intelligence research in the form of monographs, doctoral dissertations, textbooks, handbooks and proceedings volumes. The FAIA series contains several sub-series, including “Information Modelling and Knowledge Bases” and “Knowledge-Based Intelligent Engineering Systems”. It also includes the biennial ECAI, the European Conference on Artificial Intelligence, proceedings volumes, and other ECCAI – the European Coordinating Committee on Artificial Intelligence – sponsored publications. An editorial panel of internationally well-known scholars is appointed to provide a high quality selection.

Series Editors:

J. Breuker, R. Dieng-Kuntz, N. Guarino, J.N. Kok, J. Liu, R. López de Mántaras,
R. Mizoguchi, M. Musen, S.K. Pal and N. Zhong

Volume 184

Recently published in this series

- Vol. 183. C. Eschenbach and M. Grüninger (Eds.), Formal Ontology in Information Systems – Proceedings of the Fifth International Conference (FOIS 2008)
- Vol. 182. H. Fujita and I. Zualkernan (Eds.), New Trends in Software Methodologies, Tools and Techniques – Proceedings of the seventh SoMeT_08
- Vol. 181. A. Zgrzywa, K. Choroś and A. Siemiński (Eds.), New Trends in Multimedia and Network Information Systems
- Vol. 180. M. Virvou and T. Nakamura (Eds.), Knowledge-Based Software Engineering – Proceedings of the Eighth Joint Conference on Knowledge-Based Software Engineering
- Vol. 179. A. Cesta and N. Fakotakis (Eds.), STAIRS 2008 – Proceedings of the Fourth Starting AI Researchers’ Symposium
- Vol. 178. M. Ghallab et al. (Eds.), ECAI 2008 – 18th European Conference on Artificial Intelligence
- Vol. 177. C. Soares et al. (Eds.), Applications of Data Mining in E-Business and Finance
- Vol. 176. P. Zaraté et al. (Eds.), Collaborative Decision Making: Perspectives and Challenges
- Vol. 175. A. Briggles, K. Waelbers and P.A.E. Brey (Eds.), Current Issues in Computing and Philosophy
- Vol. 174. S. Borgo and L. Lesmo (Eds.), Formal Ontologies Meet Industry
- Vol. 173. A. Holst et al. (Eds.), Tenth Scandinavian Conference on Artificial Intelligence – SCAI 2008
- Vol. 172. Ph. Besnard et al. (Eds.), Computational Models of Argument – Proceedings of COMMA 2008
- Vol. 171. P. Wang et al. (Eds.), Artificial General Intelligence 2008 – Proceedings of the First AGI Conference

ISSN 0922-6389

Artificial Intelligence Research and Development

Proceedings of the 11th International Conference of
the Catalan Association for Artificial Intelligence

Edited by

Teresa Alsinet

Department of Computer Science, Universitat de Lleida, Spain

Josep Puyol-Gruart

Institut d'Investigació en Intel·ligència Artificial, CSIC, Spain

and

Carme Torras

Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Spain

IOS
Press

Amsterdam • Berlin • Oxford • Tokyo • Washington, DC

© 2008 The authors and IOS Press.

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, without prior written permission from the publisher.

ISBN 978-1-58603-925-7

Library of Congress Control Number: 2008936675

Publisher

IOS Press

Nieuwe Hemweg 6B

1013 BG Amsterdam

Netherlands

fax: +31 20 687 0019

e-mail: order@iospress.nl

Distributor in the UK and Ireland

Gazelle Books Services Ltd.

White Cross Mills

Hightown

Lancaster LA1 4XS

United Kingdom

fax: +44 1524 63232

e-mail: sales@gazellebooks.co.uk

Distributor in the USA and Canada

IOS Press, Inc.

4502 Rachael Manor Drive

Fairfax, VA 22032

USA

fax: +1 703 323 3668

e-mail: iosbooks@iospress.com

LEGAL NOTICE

The publisher is not responsible for the use which might be made of the following information.

PRINTED IN THE NETHERLANDS

Conference Organization

The CCIA 2008 conference was organized by the Associació Catalana d'Intel·ligència Artificial and the Institut d'Investigació en Intel·ligència Artificial, CSIC.

General Chair

Carme Torras, Institut de Robòtica i Informàtica Industrial, CSIC-UPC

Organizing Chair

Josep Puyol-Gruart, Institut d'Investigació en Intel·ligència Artificial, CSIC

Program Committee Chair

Teresa Alsinet, Universitat de Lleida

Scientific Committee

Núria Agell, Universitat Ramon Llull

Josep Aguilar, Laboratoire d'Architecture et d'Analyse des Systèmes, CNRS

Guillem Alenyà, Institut de Robòtica i Informàtica Industrial, CSIC-UPC

René Alquézar, Universitat Politècnica de Catalunya

Cecilio Angulo, Universitat Politècnica de Catalunya

Carlos Ansótegui, Universitat de Lleida

Eva Armengol, Institut d'Investigació en Intel·ligència Artificial, CSIC

Ester Bernardó-Mansilla, Universitat Ramon Llull

Xavier Binefa, Universitat Autònoma de Barcelona

Vicent Botti, Universidad Politécnica de Valencia

Miguel Ángel Cazorla, Universitat d'Alacant

Jesús Cerquides, Universitat de Barcelona

Carlos Iván Chesñevar, Universidad Nacional del Sur, Argentina

Ulises Cortés, Universitat Politècnica de Catalunya

Josep Lluís de la Rosa, Universitat de Girona

Teresa Escrig, Universitat Jaume I

Marc Esteva, Institut d'Investigació en Intel·ligència Artificial, CSIC

Francesc Ferri, Universitat de València

Josep Maria Garrell, Universitat Ramon Llull

Héctor Geffner, Universitat Pompeu Fabra

Lluís Godo, Institut d'Investigació en Intel·ligència Artificial, CSIC

Elisabet Golobardes, Universitat Ramon Llull

Jordi Levy, Institut d'Investigació en Intel·ligència Artificial, CSIC

Chu Min Li, Université de Picardie

Angeles López, Universitat Jaume I

Beatriz López, Universitat de Girona

Ramon López de Mántaras, Institut d'Investigació en Intel·ligència Artificial, CSIC

Maite López-Sánchez, Universitat de Barcelona

Gábor Lugosi, Universitat Pompeu Fabra
 Felip Manyà, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Rodrigo Martínez, Universidad de Murcia
 David Masip, Universitat Oberta de Catalunya
 Joaquim Meléndez, Universitat de Girona
 Violeta Migallón, Universitat d'Alacant
 Bernardo Morcego, Universitat Politècnica de Catalunya
 Antonio Moreno, Universitat Rovira i Virgili
 Pablo Noriega, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Jordi Planes, Universitat de Lleida
 Enric Plaza, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Monique Polit, Université de Perpignan 'Via Domitia'
 Oriol Pujol, Universitat de Barcelona
 Petia Radeva, Universitat Autònoma de Barcelona
 Jordi Recasens, Universitat Politècnica de Catalunya
 David Riaño, Universitat Rovira i Virgili
 Juan Antonio Rodríguez, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Horacio Rodríguez, Universitat Politècnica de Catalunya
 Ignasi Rodríguez-Roda, Universitat de Girona
 Jordi Sabater-Mir, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Miquel Sànchez-Marrè, Universitat Politècnica de Catalunya
 Sandra Sandri, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Vicenç Torra, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Aïda Valls, Universitat Rovira i Virgili
 Llorenç Valverde, Universitat de les Illes Balears
 Maria Vanrell, Universitat Autònoma de Barcelona
 Wamberto Vasconcelos, University of Aberdeen
 Glòria Vázquez, Universitat de Lleida
 Jordi Vitrià, Universitat Autònoma de Barcelona

Additional Referees

Josep Argelich, Fernando de la Rosa, Isabela Drummond, Pere Garcia-Calvés

Organizing Committee

Lluís Godo, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Eva Armengol, Institut d'Investigació en Intel·ligència Artificial, CSIC
 Joaquim Meléndez, Universitat de Girona

Organizing Institutions



Associació Catalana
d'Intel·ligència Artificial



Institut d'Investigació
en Intel·ligència Artificial

Sponsoring Institutions



This page intentionally left blank

Preface

There was a time when AI was seen by many as science fiction, i.e., the healthy endeavor of speculating about the future. Now the future is here. AI has passed from being a visionary discipline to lying at the core of many commercial enterprises. AI programs scattered through the web influence nowadays our lives: by extracting profiles and offering tailored advertisement, helping us in our searches, establishing social networks, providing entertainment... And not just in the net, but also in the physical world. In Japan there are robots that guide customers through marketplaces advising them where to find the product matching their needs, and realistic replicas of university professors allow them to teach their lectures a hundred kilometers away from the classroom. Not to speak about intelligent prostheses and remote high-precision surgery.

In the Catalan-speaking world we do not have robots in marketplaces yet, but it is coming. Recently, the first commercial humanoid robot has been built. Since AI technology is becoming reasonably mature, companies are progressively relying on it. The Catalan Association for Artificial Intelligence (ACIA¹) tries to promote synergies within the research community and also between the different actors playing a role in the development of AI: from universities to industry, from governmental departments to the information society, from entertainment enterprises to citizen services.

One of the main activities of ACIA is the organization of this annual conference (CCIA), which reaches its 11th edition here in Sant Martí d'Empúries, October 22–24, 2008. The good health of basic and applied research in the Catalan AI community and its influence area shows up in the selection of representative papers submitted to CCIA 2008, which are gathered in this volume.

The book is organized according to the different areas in which the papers were distributed for their presentation during the conference. Namely: Agents; Constraints, Satisfiability, and Search; Knowledge and Information Systems; Knowledge Representation and Logic; Machine Learning; Multidisciplinary Topics and Applications; Reasoning about Plans, Processes, and Actions; Robotics; and Uncertainty in AI. Papers appearing in this volume were subjected to rigorous blind review: two scientific committee members (or in some cases, auxiliary reviewers) reviewed each paper under the supervision of the program chairs. The scientific committee members assigned to each paper were determined based on their expertise and their expressed interest in the paper, with an eye toward coverage of the relevant aspects of each paper. This year 54 papers were submitted to CCIA, with 45 accepted for oral or poster presentation at the conference. All accepted papers appear in this volume. The quality of the papers was high in average, and the selection between oral or poster presentation was only based on the potential degree of discussion that a paper we thought could generate. We believe that all the papers collected in this volume can be of interest to any computer scientist or engineer interested in AI.

¹ ACIA, the Catalan Association for Artificial Intelligence, is a member of the European Coordinating Committee for Artificial Intelligence (ECCAI). <http://www.acia.org>.

We would like to express our sincere gratitude to all the authors and members of the scientific and organizing committees that have made this conference a success. Our special thanks go also to the plenary speakers, Ricardo Baeza-Yates and Hector Geffner, for their effort in preparing very interesting lectures, and to the president of ACIA, Núria Agell, for her kind support.

Sant Martí d'Empúries, October 2008

Teresa Alsinet, Universitat de Lleida

Josep Puyol-Gruart, Institut d'Investigació en Intel·ligència Artificial, CSIC

Carme Torras, Institut de Robòtica i Informàtica Industrial, CSIC-UPC

Contents

| | |
|--|----|
| Conference Organization | v |
| Preface | ix |
| <i>Teresa Alsinet, Josep Puyol-Gruart and Carme Torras</i> | |
| Invited Talks | |
| Web Mining or the Wisdom of the Crowds | 3 |
| <i>Ricardo Baeza-Yates</i> | |
| AI at 50: From Programs to Solvers – Models and Techniques for General Intelligence | 4 |
| <i>Hector Geffner</i> | |
| Agents | |
| Scalable and Efficient Multiagent Platform Closer to the Operating System | 7 |
| <i>Juan M. Alberola, Jose M. Such, Agustin Espinosa, Vicent Botti and Ana Garcia-Fornes</i> | |
| Designing Norms in Virtual Organizations | 16 |
| <i>E. Argente, N. Criado, V. Julián and V. Botti</i> | |
| Distributed Barter-Based Directory Services | 24 |
| <i>David Cabanillas, Steven Willmott and Ulises Cortés</i> | |
| Trading Paper Clips – An Analysis of “Trading Up” in Artificial Societies Without Altruists | 33 |
| <i>David Cabanillas, Steven Willmott and Ulises Cortés</i> | |
| Validation and Experimentation of a Tourism Recommender Agent Based on a Graded BDI Model | 41 |
| <i>Ana Casali, Lluís Godo and Carles Sierra</i> | |
| Agent Negotiation Dissolution | 51 |
| <i>Nicolás Hormazábal, Josep Lluís de la Rosa i Esteve and Silvana Aciar</i> | |
| On Partial Deduction and Conversational Agents | 60 |
| <i>Mariela Morveli-Espinoza and Josep Puyol-Gruart</i> | |
| Robustness in Recurrent Auctions for Resource Allocation | 70 |
| <i>Victor Muñoz and Didac Busquets</i> | |
| Using Electronic Institutions for Hospitals Chronic Disease Management and Purchasing System | 80 |
| <i>Ashkan Musavi, Maite Lopez-Sanchez, Jordi Campos and Marc Esteve</i> | |
| Categorization and Social Norms Support | 88 |
| <i>Daniel Villatoro and Jordi Sabater-Mir</i> | |

Constraints, Satisfiability, and Search

| | |
|--|-----|
| How Hard Is a Commercial Puzzle: The Eternity II Challenge <i>Carlos Ansótegui, Ramon Béjar, Cèsar Fernández and Carles Mateu</i> | 99 |
| Random SAT Instances à la Carte <i>Carlos Ansótegui, Maria Luisa Bonet and Jordi Levy</i> | 109 |
| Privacy in Distributed Meeting Scheduling <i>Ismel Brito and Pedro Meseguer</i> | 118 |
| Solving the Response Time Variability Problem by Means of Multi-Start and GRASP Metaheuristics <i>Albert Corominas, Alberto García-Villoria and Rafael Pastor</i> | 128 |
| An Algorithm Based on Structural Analysis for Model-Based Fault Diagnosis <i>Esteban R. Gelso, Sandra M. Castillo and Joaquim Armengol</i> | 138 |

Knowledge and Information Systems

| | |
|---|-----|
| Knowledge Discovery with Explained Case-Based Reasoning <i>Eva Armengol</i> | 151 |
| Improving Pseudobagging Techniques <i>Angela Chieppa, Karina Gibert, Ignasi Gómez-Sebastià and Miquel Sànchez-Marrè</i> | 161 |
| Knowledge Discovery on the Response to Neurorehabilitation Treatment of Patients with Traumatic Brain Injury Through an AI&Stats and Graphical Hybrid Methodology <i>Karina Gibert, Alejandro García-Rudolph, Alberto García-Molina, Teresa Roig-Rovira, Montserrat Bernabeu and José María Tormos</i> | 170 |
| Using Ensemble-Based Reasoning to Help Experts in Melanoma Diagnosis <i>Ruben Nicolas, Elisabet Golobardes, Albert Fornells, Sonia Segura, Susana Puig, Cristina Carrera, Joseph Palou and Josep Malveyh</i> | 178 |
| Ergonomic Advice Through Case-Based Reasoning to Avoid Dangerous Positions Adopted Using the Computer <i>Fernando Orduña Cabrera, Miquel Sànchez-Marrè, Jesús Miguel García Gorrostieta and Samuel González López</i> | 186 |

Knowledge Representation and Logic

| | |
|---|-----|
| Probabilistic Dynamic Belief Logic for Image and Reputation <i>Isaac Pinyol, Jordi Sabater-Mir and Pilar Dellunde</i> | 197 |
| Aggregation Operators and Ruled Surfaces <i>J. Recasens</i> | 206 |
| New Challenges: Group Decision Systems by Means of Entropy Defined Through Qualitative Reasoning Structures <i>Llorenç Roselló, Francesc Prats, Mónica Sánchez and Núria Agell</i> | 215 |

Machine Learning

| | |
|--|-----|
| Voltage Sag Source Location from Extracted Rules Using Subgroup Discovery <i>Víctor Barrera, Beatriz López, Joaquim Meléndez and Jorge Sánchez</i> | 225 |
| Statistical Monitoring of Injection Moulds <i>Xavier Berjaga, Joaquim Melendez and Alvaro Pallares</i> | 236 |
| On the Dimensions of Data Complexity Through Synthetic Data Sets <i>Núria Macià, Ester Bernadó-Mansilla and Albert Orriols-Puig</i> | 244 |
| Can Evolution Strategies Improve Learning Guidance in XCS? Design and Comparison with Genetic Algorithms Based XCS <i>Sergio Morales-Ortigosa, Albert Orriols-Puig and Ester Bernadó-Mansilla</i> | 253 |
| Intersection and Signed-Intersection Kernels for Intervals <i>Francisco J. Ruiz, Cecilio Angulo and Núria Agell</i> | 262 |

Multidisciplinary Topics and Applications

| | |
|--|-----|
| Representation of Discrete Quasi-Copulas Through Non-Square Matrices <i>Isabel Aguiló, Jaume Suñer and Joan Torrens</i> | 273 |
| A System to Extract Social Networks Based on the Processing of Information Obtained from Internet <i>Xavi Canaleta, Pablo Ros, Alex Vallejo, David Vernet and Agustín Zaballos</i> | 283 |
| A Negotiation Styles Recommenders Approach Based on Computational Ecology <i>Josep Lluís de la Rosa, Gabriel Lopardo, Nicolás Hormazábal and Miquel Montaner</i> | 293 |
| CABRO: Winner Determination Algorithm for Single-Unit Combinatorial Auctions <i>Víctor Muñoz and Javier Murillo</i> | 303 |
| Nearest Neighbor Technique and Artificial Neural Networks for Short-Term Electric Consumptions Forecast <i>Van Giang Tran, Stéphane Grieu and Monique Polit</i> | 313 |
| Pattern Discovery in Melanoma Domain Using Partitional Clustering <i>David Vernet, Ruben Nicolas, Elisabet Golobardes, Albert Fornells, Carles Garriga, Susana Puig and Josep Malvehy</i> | 323 |

Reasoning About Plans, Processes, and Actions

| | |
|---|-----|
| Using Ant Colony Systems with Pheromone Dispersion in the Traveling Salesman Problem <i>José Carlos Becceneri, Sandra Sandri and E.F. Pacheco da Luz</i> | 333 |
| Building Policies for Scrabble <i>Alejandro Gonzalez-Romero and René Alquezar</i> | 342 |

Robotics

| | |
|---|-----|
| Monocular Object Pose Computation with the Foveal-Peripheral Camera of the Humanoid Robot Armar-III <i>Guillem Alenyà and Carme Torras</i> | 355 |
| The SLAM Problem: A Survey <i>José Aulinas, Yvan Petillot, Joaquim Salvi and Xavier Lladó</i> | 363 |
| An Approach for Mail-Robot Navigation Using a CBR Technique <i>Martí Navarro, Stella Heras and Vicente Julián</i> | 372 |
| Representing Qualitative Trajectories by Autonomous Mobile Robots <i>J.C. Peris, J. Plana and M.T. Escrig</i> | 380 |
| Object-Based Place Recognition for Mobile Robots Using Panoramas <i>Arturo Ribes, Arnau Ramisa, Ramon Lopez de Mantaras and Ricardo Toledo</i> | 388 |
| Motion Segmentation: A Review <i>Luca Zappella, Xavier Lladó and Joaquim Salvi</i> | 398 |

Uncertainty in AI

| | |
|---|-----|
| On Fuzzy Description Logics <i>Àngel García-Cerdàña and Francesc Esteva</i> | 411 |
| An Interval-Based Approach for Fault Isolation and Identification in Continuous Dynamic Systems <i>Esteban R. Gelso, Sandra M. Castillo and Joaquim Armengol</i> | 421 |
| Effects of Orness and Dispersion on WOVA Sensitivity <i>Vicenç Torra</i> | 430 |
| Keyword Index | 439 |
| Author Index | 441 |

Invited Talks

This page intentionally left blank

Web Mining or The Wisdom of the Crowds

Ricardo BAEZA-YATES

Yahoo! Research

Barcelona Media Innovation Centre

ricardo.baeza@barcelonamedia.org

Web: research.yahoo.com

The Web continues to grow and evolve very fast, changing our daily lives. This activity represents the collaborative work of the millions of institutions and people that contribute content to the Web as well as the one billion people that use it. In this ocean of hyperlinked data there is explicit and implicit information and knowledge. Web Mining is the task of analyzing this data and extracting information and knowledge for many different purposes. The data comes in three main flavors: content (text, images, etc.), structure (hyperlinks) and usage (navigation, queries, etc.), implying different techniques such as text, graph or log mining. Each case reflects the wisdom of some group of people that can be used to make the Web better. For example, user generated tags in Web 2.0 sites. In this talk we walk through this process and give specific examples.

AI at 50: From Programs to Solvers

Models and Techniques for General Intelligence

Hector GEFNER

ICREA & Universitat Pompeu Fabra

Barcelona, Spain

Over the last 20 years, a significant change has occurred in AI research as many researchers have moved from the early AI paradigm of writing programs for ill-defined problems to writing solvers for well-defined mathematical models such as Constraint Satisfaction Problems, Strips Planning, SAT, Bayesian Networks and Partially Observable Markov Decision Processes. Solvers are programs that take a compact description of a particular model instance (a planning problem, a CSP instance, and so on) and automatically compute its solution. Unlike the early AI programs, solvers are general as they must deal with any problem that fits the model. This presents a crisp computationally challenge: how to make these solvers scale up to large and interesting problems given that all these models are intractable in the worst case. Work in these areas has uncovered techniques that accomplish this by automatically recognizing and exploiting the structure of the problem at hand. My goal in this talk is to articulate this research agenda, to go over some of ideas that underlie these techniques, and to show the relevance of these models and techniques to those interested in models of general intelligence and human cognition.

Agents

This page intentionally left blank

Scalable and Efficient Multiagent Platform Closer to the Operating System¹

Juan M. ALBEROLA, Jose M. SUCH², Agustin ESPINOSA,
Vicent BOTTI and Ana GARCIA-FORNES

*Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València, Camí de Vera s/n. 46022, València*

Abstract.

Agent technology demands support to build complex, open and large-scale applications. Therefore, efficient, scalable and robust Multiagent Platforms are needed. The performance of current Multiagent Platforms, which are usually developed in Java, has been tested in different studies. Results presented in these studies show that the performance of these Multiagent Platforms is not as good as one could expect. We present in this paper a different approach of a Multiagent Platform closer to the Operating System level to improve the performance achieved. This platform is focused on offering the main services required by a Multiagent platform and also improving the functionality of some of them, such as the messaging service. In this sense, we extend this functionality with advanced communication mechanisms such as agent groups and predefined interaction protocols. We also present a performance evaluation of the current platform implementation.

Keywords. Multiagent Platforms, Multiagent Systems

1. Introduction

Agent technology demands support to build complex, open and large-scale applications. In these environments, scalability, efficiency and robustness of the multiagent platform (MAP) which executes these applications become essential features. As mentioned in [8], agent researchers should design and implement large software systems, consisting of hundreds of agents, and not only composed by few agents. In order to develop these systems, researchers require efficient and scalable MAPs. Current MAP designs are not usually focused on obtaining the high performance of the MAP. Nowadays, in most MAP designs it seems that functionality takes priority over efficiency. Some research groups, for example, develop their own MAP to check some of the functionalities provided, like mobility of agents, or they build a MAP to prove theories about agent reasoning, normative systems, argumentation, etc. or they simply build a prototype of a MAP following the current trends for developing MAPs.

¹This work was supported by CONSOLIDER-INGENIO 2010 under grant CSD2007-00022 and Spanish government and FEDER funds under TIN2005-03395 and TIN2006-14630-C03-01 projects.

²Enjoying a grant of *Conselleria d'Empresa, Universitat i Ciència de la Generalitat Valenciana* (BFPI06/096).

Some current MAPs are not suitable for executing complex systems because their designs are not oriented to improve efficiency and scalability issues. Previous studies [13], [8], [20], [6], [9], [7], [12], [15], [14], [16] have analyzed some internal features of current MAPs showing that the degradation rate of both efficiency and performance increases as the system grows. Furthermore, when large-scale Multiagent Systems (MAS) are taken into account, the performance of many MAPs is considerably degraded when increasing the size of the system executed even some MAPs fail as stated in [18].

Current approaches for developing MAPs are based on interpreted languages like Java or Python. Although these languages offer some advantages like portability and easy development, MAPs developed in these languages do not perform as well as one might expect, especially when we are running complex systems.

This behaviour can be influenced by middlewares like the Java Virtual Machine (JVM) required by interpreted languages. Previous work led us to the conclusion that using current approaches to develop MAPs affects the performance of the system being executed. We proved that using the operating system (OS) services to develop a MAP instead of using middlewares between the OS and the MAP noticeably improves the performance and scalability of the MAP.

As far as we are concerned, MAP functionality can be seen as an extension of the functionality offered by the OS. Therefore, robustness and efficiency of MAPs should be required, just as we require the robustness and efficiency of an OS. In our opinion, developing MAPs closer to the OS may improve the performance of complex, open and large-scale MAS due to the fact that no middleware, such as JVM, is required. This design can increase robustness, efficiency and scalability.

Furthermore, bringing MAP design closer to the OS level, allows us to define a long term objective, i.e., to incorporate the agent concept to the OS itself to offer a greater abstraction level than current approaches. Following this trend, our research group is developing a MAP closer to the OS level.

We propose a new approach for developing MAS by means of a MAP closer to the OS level. This MAP is not only aimed at offering functionality for developing MAS but is also oriented to perform at a high level, even when we are executing large-scale systems, e. g., massively MAS that are composed of thousands or millions of agents running on several hosts. In these scenarios, features like efficiency and scalability are especially required.

This MAP is presented in this paper. It has been developed using the services offered by the OS to support MAS efficiently. If we can reach our goal of bringing closer the functionality of a MAP to the OS level, we will be able to incorporate this functionality into the OS itself in the future, with the intrinsic advantages provided by working at OS level, such as security, efficiency and high performance.

The rest of the article is organized as follows. Section 2 presents the internal architecture of this MAP. Section 3 details the services offered by this MAP. Section 4 explain the internal structure of an agent of this MAP. In section 5 we evaluate the performance of this MAP versus a Java-based platform. Finally, section 6 presents some concluding remarks.

2. Magentix Multiagent Platform architecture

Most current MAPs do not take into account features such as efficiency or scalability when these are designed. Actually, some MAPs can fail when they are launched on more than five hosts or their performance is notably downgraded, relative to the growth of the MAP load. We consider that efficiency of a MAP is essential, especially if we want to run complex MAS. For this reason, in order to develop our MAP, we checked all design alternatives offered by OS services. We also empirically analysed every design decision taken and finally we selected the best option to design a scalable and efficient MAP.

The MAP presented in this paper is called Magentix multiagent platform (a MultiAGENT platform integrated into LINUX) [3]. To offer a response time with taxes closer to the achievable time lower bound, we have to develop this MAP using the services offered by the OS. Thus, one of the design decisions is that this MAP is written in C over the Linux OS. Firstly we focus our efforts on the development of a high performance communication service. This service is quite crucial to the performance of the MAP and some other services may be implemented using it. Magentix provides advanced communication mechanisms such as agent groups, predefined interaction protocols and a security mechanism to provide authentication, integrity, confidentiality and access control. This design has been developed in order to offer the functionality required by MAS and performing efficiently.

Magentix is a distributed MAP composed of a set of computers executing Linux OS (figure 1). Magentix uses replicated information on each MAP host to achieve better efficiency. Each one of these computers presents a process tree structure. Using the advantage of process trees management offered by Linux, and using some services like signals, shared memory, execution threads, sockets, etc. we have a suitable scenario for developing a robust, efficient and scalable MAP.

Structure of each Magentix host is a three-level process tree. On the higher level we see the *magentix* process. This process is the first one launched on any host when it is added to the MAP. Below this level we can see the services level. Magentix provides some services like the *AMS* or *DF* defined by FIPA [1],[10]. These services support agent execution and are represented by means of processes replicated in every MAP host (called *ams*, *df*, etc.). Processes representing the same service communicate with each other, share and exchange information by means of replication mechanisms.

As we can see, *magentix* process is the father process of each process associated with a service. Using a process tree structure, *magentix* process manages its child processes completely (processes associated to services). By means of signal sending, *magentix* detects at once when a child process has died. Furthermore, *magentix* process can kill a service process sending a signal. This automated control provided by the relation between the father and child process allows us to manage any time a service process is created or dies. We achieve to shutdown the MAP in a controlled way and we are able to relaunch a service any time it fails without using complex and performance influent mechanisms like periodic checking requests.

Finally, in the third process level, agents are placed. Each Magentix agent is represented by a different Linux child process of the *ams* process running on the same hosts. In the following sections we will see this relationship in more detail.

We need to provide communication for agent interactions, agent and service interactions and probably some others. In an OS context there are different alternatives avail-

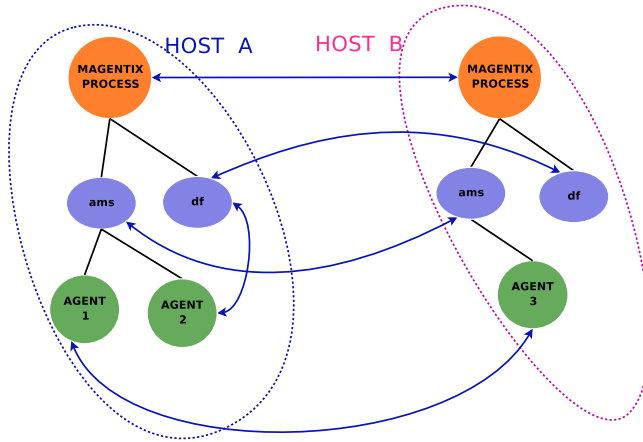


Figure 1. Platform structure

able for communicating processes. In previous studies we have analyzed different communication services among processes provided by POSIX [4] compliant OS, especially, the Linux OS, in order to select which of these services allow building robust, efficient and scalable MAPs. In these studies, we showed to what extent the performance of a Message Transport System (MTS) degrades when its services are based on middlewares between the OS and the MAP (like the JVM) rather than directly by the underlying OS.

At a lower level, Java-RMI technology (used for development communication in most of MAPs based on Java) uses TCP sockets. As a consequence, the overhead resulting from the use of Java-RMI can be avoided using TCP sockets directly, because the agent abstraction provided by a MAP is independent of the underlying communication mechanism implementation. In other words, agent developers do not deal with either Java-RMI or TCP sockets. After evaluating different alternatives, we finally define the communication mechanism implemented in Magentix as point to point connections based on TCP sockets, between a pair of processes [5].

Distributed architecture of Magentix uses replicated information in each MAP host for achieving better efficiency. Most of the information concerning the MAP, agents, services, etc. needs to be replicated in each host.

3. Services

In this section we describe two of the distributed services that are implemented in Magentix: *AMS* service and *DF* service.

3.1. AMS (Agent Management System)

AMS service offers the white pages functionality defined by FIPA. This service stores the information regarding the agents that are running on the MAP. This service is distributed among every MAP host. As we previously said, each host runs an *ams* process that is a child process of the *magentix* process. The *ams* processes share information with each other regarding the agents of the MAP.

As we said in section 2, all the agents launched in a specific host are represented by means of child processes of the *ams* process. Just as the *magentix* process behaves, the *ams* process has a broad control of the agents of this host. So, every time a new agent starts its execution or finishes, the *ams* process automatically manages these events by means of sending signals.

The *AMS* service stores the information regarding every agent running on the MAP. This service allows us to obtain the physical address (IP address and port) providing the agent name to communicate with. Due to the fact that the *AMS* service is distributed among every MAP host, each *ams* process contains the information needed for contacting every agent of the MAP. Hence, the operation of searching agent addresses is not a bottleneck as each agent looks this information up in its own host, without needing to make any requests to centralizing components.

Changes that need to be known in every host are for instance the creation or deletion of agents. Nevertheless, other information exists regarding agents that do not need to be replicated. For this reason, the *ams* processes manages two tables of information: the Global Agent Table (*GAT*) and the Local Agent Table (*LAT*).

- **GAT:** In this table is stored the name of each agent of the MAP and its physical address, that is, its IP address and is associated port.
- **LAT:** In this table additional information is stored such as the agent owner, the process PID that represents each agent and its life cycle state.

The information contained in the *GAT* needs to be replicated in each host to achieve better performance. The *GAT* is mapped on shared memory. Every agent has read only access to the information stored in the *GAT* of its same host. Thus, each time an agent needs to obtain the address of another agent to communicate, it accesses to the *GAT* without making any request to the *ams* process. Thus, we avoid the bottleneck of requesting centralizing components each time one agent wants to communicate to another.

On the other hand, the information of the *LAT* is not replicated. Some information stored in the *LAT* regarding a specific agent is only needed by the *ams* process of the same host (for instance the process PID), so it does not need to be replicated in each host. Some other information could be useful for the agents but is usually not requested (such as the life cycle state). Thus, to reduce the overhead introduced by replication, we divide the information regarding agents in two tables. Therefore, each *ams* stores in their *LAT* the information regarding the agents under its management, that is, the agents that are running on the same host. If some information available to agents is needed (such as the life cycle state), the agent has to make up a request to the specific *ams*.

3.2. *DF* (Directory Facilitator)

The *DF* service offers the yellow pages functionality defined by FIPA. This service stores the information regarding the services offered by MAP agents. The *DF* service allows agents to register services offered by them, deregister these services or look up some specific service required. Similar to the *AMS* service, the *DF* service is implemented in a distributed scenario by means of processes running on each MAP host, called *df*. In every MAP host exists a single *df* process, which is child process of the *magentix* process. Information regarding services offered by MAP agents is replicated in every *df* process of the MAP.

The information which needs to be replicated is stored in a unique table called **GST** (Global Service Table). This table is a list of pairs: services offered by agents of the MAP and the agent that offers this service. In contrast to the **GAT**, **GST** is not implemented as shared memory, so only the *df* can directly access this information.

Agents can register, deregister or look up services offered by other agents. To do these tasks, agents need to communicate with the *df* process of its own host. Therefore, each *df* has an interface to communicate with agents by message exchange.

3.3. Advanced Communication Functionalities

Another available functionality offered by Magentix are the predefined interaction protocols. Interaction protocols defined by FIPA are implemented in Magentix. User can implement conversations among agents using these standard protocols. We provide an API to manage these protocols and the user can easily adapt them according their needs. We also define conversation processors that users can associate with different agent conversations. The user can associate a processor with different interaction protocols and this processor will manage every agent conversation that executes some of these protocols. Using this functionality users can easily manage conversations among agents.

Another functionality offered by Magentix is the agent group. Users can define a group of agents as an abstract entity for offering some service or just for managing messages. Using this approach we can communicate not only a pair of agents but also a single agent to a group. Magentix offers a Organizational Unit Manager (*OUM*) to manage agent groups defined as organizational units. Every unit can receive messages as a single agent, addressing these messages to one, several or all the agents of the unit. We provide a routing mechanism to deliver messages to units. Using this concept, agents see a unit as a single agent but they may be composed of several agents.

We also provide a security mechanism which offers authentication, integrity, confidentiality based on Kerberos and an access control mechanism using services provided by Linux OS [17].

4. Magentix Agents

Agents in Magentix are represented as Linux processes. Internally, every agent is composed by Linux threads: a single thread for executing the agent tasks (main thread), a thread for sending messages (sender thread) and a thread for receiving messages (receiver thread). Each agent process is a child process of the *ams* running on the same host. This *ams* manages the creation and deletion of agents running on the same host. The *ams* GAT is shared between an *ams* process and the agents, so accessing the physical addresses of any agent of the MAP is fast and does not become a bottleneck.

An agent communicates to other agents and services by means of message sending. We provide an API to manage these operations. We also provide an extended API for managing messages according the FIPA ACL Message Structure Specification [11]. This message structure allows users to easily manage every sent message and received message, and to process conversations between different agents.

Internal communication mechanism is carried out by means of point to point connections. These connections are created and removed by internal agent threads. Each agent

is associated with a specific pair (IP address & port) for opening TCP connections to exchange messages with other agents or services. A connection between a pair of agents represents a conversation. The first time an agent contacts another one, a TCP connection is established and remains open to exchange messages in the future. Each agent stores its open connections in a *connection table*. These connections are automatically closed when the conversation is not active, that is, some time is spent since last sent message according a LRU policy. This mechanism is more detailed described in previous studies [19].

5. Performance Evaluation

In order to evaluate the behaviour of this new MAP approach we present a preliminary performance study. We carry out an experiment in order to measure the response time of the MAP. We propose to launch a sender agent that exchanges messages of 10 bytes with a receiver agent. We place these two agents on different hosts and we are increasing the number of couples. We measure the elapsed time from when all the couples begin (in a synchronized way) exchanging messages to when all the couples have exchanged 1000 messages.

The whole set of experiments presented in this paper have been performed using 2 pc Intel(R) Pentium(R) 4 CPU 3.00GHz, with 1GB of RAM memory, and running the Ubuntu Linux 6.06 OS (kernel 2.6.15).

We run the experiment above mentioned in Magentix platform and also in Jade MAP, that is a well-known and widely-used Java-base platform [2]. As stated in the study presented in [13], Jade presents a good messaging service, showing good response times and acceptable scalability. Jade is implemented in Java, so that it runs on top of the JVM. Jade uses Java-RMI to carry out the communication when agents are located in different hosts.

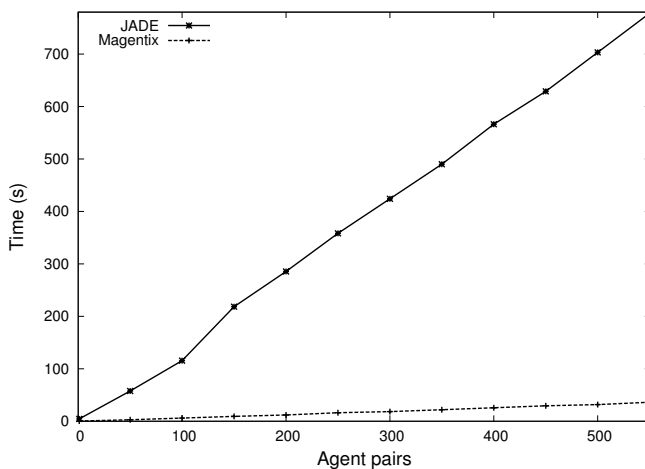


Figure 2. Magentix vs Jade platforms

In figure 2, we show the behaviour of Magentix and Jade MAPs when running this experiment. We show the response time (in seconds) of both platforms. We observe that Magentix offers a good performance. We can also see that the difference of performance between the two platforms is increased when both the number of agents and the message traffic are increased too, so Magentix is more scalable than Jade. We need to carry out an in-depth performance study measuring more parameters and comparing more MAPs, but this experiment is a first step to assess our MAP performance.

6. Conclusion

In this paper a MAP designed using a new approach is presented. Current MAPs are not usually designed taking into account the efficiency they can offer. Most of these designs prevent MAPs from being scalable and efficient. So, they are not suitable for developing large-scale systems such as massively MAS, because the performance of these MAPs becomes quite degraded if the system gets too complex or the MAP just fails.

We propose a MAP design closer to the OS level, so we try to improve the performance of the MAP, implementing it using the services offered by the OS itself, without requiring a middleware. Deleting this middleware between the OS and the MAP we have managed to noticeably improve the efficiency of the MAP, especially when running complex systems. Furthermore, in order to achieve an efficient design, we have also analyzed the different alternatives offered by the OS services to develop the MAP, and we have compared them empirically in order to select the best scalable and efficient option. We have shown empirically that the approach for designing this MAP is quite efficient and scalable, compared to a classic approach of MAPs implemented using interpreted languages. Magentix reach a high performance even when we are running large-scale systems.

We have focused on outperforming the messaging service of the MAP. We propose a basic functionality to communicate agents and we define basic services defined by FIPA as the *AMS* service and *DF* service. Moreover, we also add advanced communication mechanisms like agent groups, predefined interaction protocols and a security mechanism to provide authentication, integrity, confidentiality and access control. In the future we are planning to add functionalities related to agent-internal issues, coordination mechanisms, content languages to specify ontologies, tracing mechanisms, etc. but always taking into account efficiency and performance issues.

References

- [1] Fipa (the foundation for intelligent physical agents). <http://www.fipa.org/>.
- [2] Java agent development framework (jade). <http://jade.tilab.com>.
- [3] Magentix (multiagent platform integrated in linux). <http://www.dsic.upv.es/users/ia/sma/tools/Magentix/index.html>.
- [4] *Standard for information technology - portable operating system interface (POSIX)*.
- [5] J. M. Alberola, L. Mulet, J. M. Such, A. Garcia-Fornes, A. Espinosa, and V. Botti. Operating system aware multiagent platform design. In *Proceedings of the Fifth European Workshop on Multi-Agent Systems (EUMAS-2007)*, pages 658–667, 2007.
- [6] K. Burbeck, D. Garpe, and S. Nadjm-Tehrani. Scale-up and performance studies of three agent platforms. In *IPCCC 2004*.

- [7] D. Camacho, R. Aler, C. Castro, and J. M. Molina. Performance evaluation of zeus, jade, and skeleton-agent frameworks. In *Systems, Man and Cybernetics, 2002 IEEE International Conference on*.
- [8] K. Chmiel, D. Tomiak, M. Gawinecki, and P. Karczmarek. Testing the efficiency of jade agent platform. In *Proceedings of the ISPD/HeteroPar'04*, 49-56.
- [9] E. Cortese, F.Quarta, and G. Vitaglione. Scalability and performance of jade message transport system. *EXP*, 3:52–65, 2003.
- [10] FIPA. *FIPA Abstract Architecture Specification*. FIPA, 2001.
- [11] FIPA. *FIPA ACL Message Structure Specification*. FIPA, 2001.
- [12] L. Lee, Nwana, H.S., Ndumu, D.T., and P. De Wilde. The stability, scalability and performance of multi-agent systems. In *BT Technology J.*, volume 16, 1998.
- [13] L. Mulet, J. M. Such, J. M. Alberola, V. Botti, A. Espinosa, A. Garcia-Fornes, and A. Terrasa. Performance evaluation of open-source multiagent platforms. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS06)*, pages 1107–1109. Association for Computing Machinery, Inc. (ACM Press), 2006.
- [14] E. Shakshuki. A methodology for evaluating agent toolkits. In *ITCC '05: Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'05) - Volume I*, pages 391–396, Washington, DC, USA, 2005. IEEE Computer Society.
- [15] E. Shakshuki and Y. Jun. Multi-agent development toolkits: An evaluation. *Lecture Notes in Computer Science*, 3029:209–218, 2004.
- [16] D. Sislak, M. Rehak, M. Pechouce, M. Rollo, and D. Pavlicek. A-globe: Agent development platform with inaccessibility and mobility support, 2005.
- [17] J. M. Such, J. M. Alberola, A. García-Fornes, A. Espinosa, and V. Botti. Kerberos-based secure multi-agent platform. In *Sixth International Workshop on Programming Multi-Agent Systems (ProMAS'08)*, pages 173–186, 2008.
- [18] J. M. Such, J. M. Alberola, L. Mulet, A. Espinosa, A. Garcia-Fornes, and V. Botti. Large-scale multi-agent platform benchmarks. In *Languages, methodologies and Development tools for multi-agent systems (LADS 2007). Proceedings of the Multi-Agent Logics, Languages, and Organisations - Federated Workshops.*, pages 192–204, 2007.
- [19] J. M. Such, J. M. Alberola, L. Mulet, A. Garcia-Fornes, A. Espinosa, and V. Botti. Hacia el diseño de plataformas multiagente cercanas al sistema operativo. In *International workshop on practical applications on agents and multi-agent systems*, 2007.
- [20] P. Vrba. Java-based agent platform evaluation. In *Proceedings of the HoloMAS 2003*, pages 47–58, 2003.

Designing Norms in Virtual Organizations

E. Argente and N. Criado and V. Julián and V. Botti¹

Grupo de Tecnología Informática - Inteligencia Artificial

Departamento de Sistemas Informáticos y Computación

Universidad Politécnica de Valencia

{eargente,ncriado,vinglada,vbotti}@dsic.upv.es

Abstract. Virtual Organizations have been recently employed as an abstraction for modelling dynamical and complex systems. The Norm concept has arisen inside the Multi-Agent Theory for avoiding conflicts and ensuring coordination and cooperation in open environments. In this work, a desing of norms that govern Virtual Organizations is proposed, which is based on a service-oriented approach. Therefore, we define a formal normative language for mainly controlling service registering and usage. Thus, a better integration of both Web Services and MAS Technology is achieved.

Keywords. Multi-Agent Systems, Normative Systems, Normative Language, Normative Implementation, Virtual Organizations.

Introduction

Recently, research in Multi-Agent Systems (MAS) has focused on covering distributed, complex and dynamic systems, in which information and resources are distributed among several agents [15]. Inside these societies, agents are heterogeneous and might not share a common goal. Therefore, normative concepts have arisen inside MAS as a mechanism for avoiding conflicts and ensuring social order.

Virtual Organizations represent a set of individuals and institutions that need to coordinate resources and services across institutional boundaries [11]. They are open systems, in the sense that they are inhabited by heterogeneous agents whose behaviours are unknown to the system designer [2]. Virtual Organizations have been successfully employed as a paradigm for developing agent systems [10]. Norms prescribe what is permitted and forbidden in these societies. Thus, they define benefits and responsibilities of their members and, as a consequence, agents are able to plan their actions according to the expected behaviour. In general, any process that requires coordination and cooperation among several agents is done through the definition of norms that control this interaction.

This work proposes a normative language for regulating Virtual Organization mainly based on a service-oriented approach. This paper is structured as follows:

¹This work has been partially funded by TIN2005-03395 and TIN2006-14630-C03-01 projects of the Spanish goverment, FEDER funds and CONSOLIDER-INGENIO 2010 under grant CSD2007-00022.

next section shows an overview of related works; section 2 contains an explanation of the developed model; in section 3 the proposed normative language is described; section 4 contains a case study modelled with our language, and finally conclusions and future works are detailed in section 5.

1. Related Work

Norms have been studied from different perspectives such as philosophy, psychology, law, etc. Agent Theory also deals with normative concepts as a mechanism for ensuring coordination inside Virtual Organizations.

Initial works focused on including aspects related to norms in the agent behaviour. They took Legal Theory as a starting point, which had traditionally been studying norms [6]. These early works employed legal results in order to develop *normative autonomous agents* [7]. This kind of agents are defined as agents capable of deliberating using norms, so they can decide whether to follow a norm or even solve norm conflicts. This notion of normative agents raises several open questions that later works have tried to deal with.

MAS research has given different meanings to the norm concept. It has been employed as a synonym of obligation and authorization [7], social law [16], social commitment [20] and other kinds of rules imposed by societies or authorities. These works allow identifying norms that control a society. However, a specific mechanism for representing these concepts is needed. Several approaches have studied the employment of logics in order to formalize normative MAS [7,18].

The existence of a logic formalism that allows representing normative concepts provides agents with the possibility of deliberating over norm convenience. In some cases, the “intelligent” violation of a norm might be interesting. These works consider norms as mental attitudes [4] and try to incorporate normative reasoning into agent architectures [9,5]. Recently, some experimental works on norm adoption [13] and emergence [17] have been done.

Nowadays, several approaches take into account a more pragmatic and computational notion of norm. These approaches give a computational interpretation of norms and show how a normative theory can be included in the design and execution of real MAS. They are based on the *normative system* concept [3], i.e. a MAS in which agent behaviours are restricted by different kinds of norms. According to this, different systems that employ norms for achieving coordination among their members have been proposed [14,21]. Our approach is based on a previous work for implementing norms in Electronic Institutions [12]. Our normative language extends this previous work since it offers a more general and expressive formalism for representing norms that govern Virtual Organizations, based on a service-oriented point-of-view.

2. Virtual Organization Model

A detailed explanation of the *Virtual Organizational Model* (VOM), on which our proposed language is based, is over the scope of this article. However, a full

description of the developed model can be found in [1]. Figure 1a shows a diagram that contains the main concepts of our model and the relationships among them.

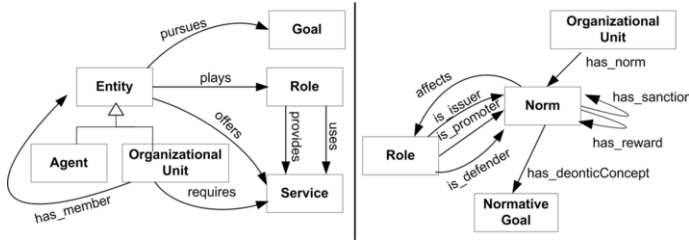


Figure 1. a)Virtual Organization Model b)Normative model

The proposed model defines an *Organizational Unit* (OU) as a basic social entity that represents the minimum set of agents that carry out some specific and differentiated activities or tasks, following a predefined pattern of cooperation and communication. This association can also be seen as a single *entity*, since it pursues *goals*, offers and requests *services* and it plays a specific *role* when it interacts with other entities. An OU is formed by different *entities* (*has_member* relationship) along its life cycle which can be both single agents or other units viewed as a single entity. On the other hand, *entities* are capable of *offering* some specific functionality. Their behaviour is motivated by their pursued goals. Moreover, an organizational unit can also publish its *requirements* of services, so then external agents can decide whether participate inside, providing these services. Any service has one or more roles that are in charge of its provision and others that consume it. Services represent the functionality that agents offer to other entities, independently of the concrete agent that makes use of it. Thus, in this proposed model MAS are modelled as service-oriented architectures.

After this overview of VOM, next section describes our normative proposal for controlling and regulating agent behaviours.

3. Normative Framework

The proposed Normative Framework describes organizational norms and normative goals that agents must follow, including sanctions and rewards.

Open systems, in which heterogeneous and self-interested agents work together, need some mechanisms that restrict agent behaviour but maintain agent autonomy. With this aim, the proposed model has been extended with a normative framework. It is a coordination mechanism that attempts to: (i) promote behaviours that are satisfactory to the organization, i.e. actions that contribute to the achievement of global goals; and (ii) avoid harmful actions, i.e. actions that prompt the system to be unsatisfactory or unstable. More specifically, norms determine role functionality by restricting its actions (prohibitions, obligations and permissions) and establishing its consequences (sanctions and rewards). Moreover, norms could be employed for defining minimum and maximum role cardinalities and also incompatibility restrictions between roles.

| | | |
|---|-----|--|
| <code><ext_norm></code> | ::= | <code><norm></code> [SANCTION(<code><norm></code>)] [REWARD(<code><norm></code>)] |
| <code><norm></code> | ::= | <code><deontic_concept></code> <code><entity></code> <code><service_action></code> [<code><temporal_situation></code>] [IF <code><if_condition></code>] <code>norm_id</code> |
| <code><deontic_concept></code> | ::= | OBLIGED FORBIDDEN PERMITTED |
| <code><entity></code> | ::= | agent: role – unit role – unit <code>entity_id</code> |
| <code><agent></code> | ::= | ? <i>variable</i> <code>agent_id</code> |
| <code><role></code> | ::= | ? <i>variable</i> <code>role_id</code> |
| <code><unit></code> | ::= | ? <i>variable</i> <code>unit_id</code> |
| <code><entity_id></code> | ::= | <code>agent_id</code> <code>role_id</code> <code>unit_id</code> |
| <code><temporal_situation></code> | ::= | BEFORE <code><situation></code> AFTER <code><situation></code> BETWEEN(<code><situation></code> , <code><situation></code>) |

Table 1. Norms BNF syntax

In our model, agents are assumed to be normative and autonomous. Hence, agents are informed of their addressed norms and they are able to reason about norm adoption. Norms allow the system to face with illicit or incorrect behaviours by performing sanctions. On the other hand, agents which carry out good actions are rewarded. Therefore, norms are not impositions; they are a method for persuading normative agents to behave correctly.

As seen in Figure 1b, each norm *affects* a role. The role is *obliged*, *forbidden* or *permitted* to perform an action. *Sanctions* and *rewards* are expressed through norms. The *issuer* role is in charge of the norm fulfilment, whereas *defender* and *promoter* roles are responsible for sanctions and rewards, respectively.

We have defined three types of norms: (i) role incompatibility norms, i.e. if an agent is playing X role it must not play Y role at the same time; (ii) role cardinality norms, that specify the minimum and maximum quantity of agents that play the same role concurrently; and (iii) functionality norms, that define role functionality in terms of services that can be requested/provided, service requesting order, service conditions, protocols that should be followed, etc. More specifically, functionality norms can be unconditional or conditional. Unconditional norms define which services must be unconditionally provided or requested by an agent and which ones must not. On the other hand, a conditional norm defines the set of actions that an agent should perform according to: previous service result, environmental states, etc.

3.1. Normative Language Description

In our model, actions are related to services. Consequently, our normative language makes possible the definition of constraints on agent behaviours in terms of actions concerning services. This normative language is based on previous works for implementing norms inside Electronic Institutions [12].

Table 1 details the reduced BNF syntax of this language. In our approach a norm is defined by means of a deontic concept (`<deontic_concept>`), an addressed role (`<entity>`) and an action (`<service_action>`). The `<temporal_situation>` establishes a temporal condition for the activation of the norm, related with other services. In addition, it can contain a state condition for the activation of the norm (`<if_condition>`). These conditions are defined in Table 4.

| | | |
|--|-----|--|
| <code><service_action></code> | ::= | <code><serv_publication> <serv_provision> <serv_usage></code> |
| <code><serv_publication></code> | ::= | <code>REGISTER <i>service_name</i> PROFILE <profile_desc></code> <code>[PROCESS<process_desc>]</code> |
| <code><service_provision></code> | ::= | <code>SERVE <i>service_name</i> PROCESS <process_desc></code> <code>[MESSAGE(<msg_cont>)]</code> |
| <code><service_usage></code> | ::= | <code>REQUEST <i>service_name</i> MESSAGE(<msg_cont>)</code> |

Table 2. Actions BNF syntax

| | | |
|-----------------------------------|-----|--|
| <code><profile_desc></code> | ::= | <code>[INPUT(<param_list>)] [OUTPUT (<param_list>)]</code> <code>[PRE(<cond_exp>)][POST(<cond_exp>)] <i>profile_id</i></code> |
| <code><process_desc></code> | ::= | <code><i>process_id</i> ?<i>variable</i> </code> <code><action> CONNECTS <process_desc> </code> <code><action> JOIN <process_desc> </code> <code>IF <cond_exp> THEN(<process_desc>)</code> <code>[ELSE (<process_desc>)] </code> <code>WHILE <cond_exp> DO(<process_desc>)</code> |
| <code><msg_cont></code> | ::= | <code>[SENDER(<entity>)] [RECEIVER (<entity>)]</code> <code>[PERFORMATIVE (<i>performative_id</i>)]</code> <code>CONTENT (<args>)</code> |
| <code><action></code> | ::= | <code><i>task_id</i>(<param_list>) <service_usage></code> |
| <code><param_list></code> | ::= | <code><i>variable</i> : <i>type</i> [,param_list]</code> |

Table 3. Service profile and process syntax

As stated before, our model actions are related to the publication (REGISTER), provision (SERVE) or usage (REQUEST) of services. The BNF syntax of actions is detailed in Table 2. Registering a new service implies that a new service profile (`<profile_desc>`) is given. Optionally, a description of the process performed by the service (`<process_desc>`) could also be provided.

A service profile contains input and output parameters of the service and its preconditions and postconditions. On the other hand, a process description details the sequence of actions (`<action>`) carried out by the service. When an agent requests a service it should send a message in whose content (`<msg_cont>`) all requesting parameters are specified. Syntax of service processes, profiles and requesting messages are contained in Table 3.

The state condition (Table 4) for the activation of the norm (`<if_condition>`) is a boolean condition that can be expressed over some variables, identifiers, normative results (`<normative_result>`) or service results (`<service_result>`). Valid normative results describe failed or satisfactory states. A `<situation>` is employed in the definition of a temporal condition. It can be expressed as a deadline, an action or a service result.

In this section a new normative framework for ensuring social order inside Virtual Organizations has been presented. It proposes a more general and expressive formalism for representing norms that govern Virtual Organizations, mainly focused on controlling service registering and usage. The definition of norms in terms of actions related to services makes possible a better integration of both MAS and Web Service technologies. Following, a case study of a normative Virtual Organization is detailed.

| | | |
|---------------------------------------|-----|--|
| <code><if_condition></code> | ::= | NOT(<cond_exp>) < cond_exp > |
| <code><cond_exp></code> | ::= | <condition> NOT <condition> <condition> AND <cond_exp> <condition> OR <cond_exp> |
| <code><condition></code> | ::= | <variant> <operator> <variant> <normative_result>(<norm>) |
| <code><normative_result></code> | ::= | FAILED SATISFIED |
| <code><operator></code> | ::= | > < >= <= = \subseteq |
| <code><variant></code> | ::= | <atomic> <formula> <i>value</i> |
| <code><atomic></code> | ::= | <i>identifier variable</i> <i>variable</i> <i>identifier</i> |
| <code><formula></code> | ::= | <i>identifier</i> (<args>) <service_result> QUANTITY(<args>) |
| <code><args></code> | ::= | <variant> [, <variant>] |
| <code><situation></code> | ::= | <i>deadline</i> <service_result> [<i>entity</i>] < service_action> |
| <code><service_result></code> | ::= | RESULT(<i>service_name</i> , <entity>, <i>ContextTime</i>) |

Table 4. Conditions syntax

4. Case Study

In this section, an example of a travel agency system has been developed employing VOM. The system acts as a regulated meeting-point, in which providers and clients contact between them for consuming or producing touristic services. A similar example has previously been modelled using electronic institutions [19,8].

The TravelAgency application facilitates the interconnection between clients (individuals, companies, travel agencies) and providers (hotel chains, airlines). It has been developed as a service oriented architecture approach, that controls and delimits which services agents can offer and/or provide. Internal functionality of these services is delegated to agent providers. However, the system imposes some restrictions on service profiles, service requesting order and service results. These restrictions are expressed by means of norms.

An organization corresponds to groups of agents that exist in the system. These agent associations are defined employing the OU concept. This case study is modelled as a main unit (*TravelUnit*). Three kinds of roles have been identified: (i) *customer*, which requests services of the system; (ii) *provider*, that offers hotel or flight search services, and might also provide hotel room or flight seat bookings; and (iii) *payee*, that gives advanced payment services and represents bank institutions. Both *customer* and *provider* are external roles, i.e. roles that can be enacted by any agent (whose functionality might have not been implemented by the system designers), so they must be controlled by the system norms and services. On the contrary, *payee* is an internal role that is assigned to internal agents of the system platform, because the payment service should be a fixed and trusted functionality.

4.1. Normative Description

In the proposed example, there are two kinds of services: general and domain-related services. The former are services or actions that agents can request to the system in order to play a role, access to some resource, etc. The domain-related

services are specific ones for the TravelAgency system: *TravelSearch*, *Reserve* and *Payment* services.

Following, examples of norms for the TravelAgency example are detailed:

- **Role Incompatibility Norms.**

- * *Payee* role is not compatible with *Provider* and *Customer* roles.

FORBIDDEN Payee REQUEST AcquireRole MESSAGE (CONTENT (role
"Provider", unit "TravelAgency") OR CONTENT (role "Customer", unit
"TravelAgency"))

- * *Customer* role is incompatible with *Provider*.

FORBIDDEN Customer REQUEST AcquireRole MESSAGE (CONTENT (role
"Provider", unit "TravelAgency"))

- **Role Cardinality Norms.**

- * *Customer* role cardinality=(0,20). Maximum role cardinality is ensured through the following norm:

FORBIDDEN Member REQUEST AcquireRole MESSAGE (CONTENT (role
"Customer", unit "TravelAgency") IF QUANTITY(Customer)≥20

- * *Provider* role cardinality=(3,∞). Minimum cardinality restriction of *Provider* role has been modelled by means of a single norm. This norm forbids agents to request *TravelSearch* service if the number of providers is lower than 3:

FORBIDDEN Member REQUEST TravelSearch IF QUANTITY(Provider)≤3

- **Functionality Norms.**

1. **Unconditional Functionality.**

- * *Customer* role cannot register any service; it is only allowed to request services.

FORBIDDEN Customer REGISTER TravelSearch PROFILE
TravelSearchProfile
FORBIDDEN Customer REGISTER Reserve PROFILE ReserveProfile

2. **Conditional Functionality.**

- * *Customer* role cannot request *Reserve* service for a specific information, unless it searches this information previously.

OBLIGED Customer REQUEST TravelSearch
MESSAGE (CONTENT (city ?c, country ?co)) BEFORE
Customer REQUEST Reserve MESSAGE(CONTENT(company ?cm city ?c))

5. Conclusions and Future Work

Norms have been employed in Agent Organizations as a mechanism for ensuring social order. In this paper, a new formal normative language for designing norms in Virtual Organizations has been presented. It proposes a more general and expressive formalism for representing norms that govern Virtual Organizations,

mainly focused on controlling service registering and usage. The definition of norms in terms of actions related to services makes possible a better integration of both MAS and Web Service Technologies.

As future work, the proposed normative framework will be implemented in a multi-agent platform. Our aim is to integrate this Virtual Organizational Model into a MAS platform. All this work belongs to a bigger project whose driving goal is to develop a new way of working with open agent organizations.

References

- [1] E. Argente, V. Julian, and V. Botti. Mas modelling based on organizations. In *Proc. AOSE08*, pages 1–12, 2008.
- [2] G. Boella, J. Hulstijn, and L. W. N. van der Torre. Virtual organizations as normative multiagent systems. In *Proc. HICSS*. IEEE Computer Society, 2005.
- [3] G. Boella and L. van der Torre. Regulative and constitutive norms in normative multiagent systems. In *KR'04*, pages 255–265. AAAI Press, 2004.
- [4] C. Castelfranchi. Prescribed mental attitudes in goal-adoption and norm-adoption. *Artif. Intell. Law*, 7(1):37–50, 1999.
- [5] C. Castelfranchi, F. Dignum, C. M. Jonker, and J. Treur. Deliberative normative agents: Principles and architecture. In *ATA, LLNCS 1757*, pages 364–378, 1999.
- [6] R. Conte, R. Falcone, and G. Sartor. Introduction: Agents and norms: How to fill the gap? *Artif. Intell. Law*, 7(1):1–15, 1999.
- [7] F. Dignum. Autonomous agents with norms. *Artif. Intell. Law*, 7(1):69–79, 1999.
- [8] F. Dignum, V. Dignum, J. Thangarajah, L. Padgham, and M. Winikoff. Open Agent Systems?? In *AOSE*, 2007.
- [9] F. Dignum, D. Morley, L. Sonenberg, and L. Cavedon. Towards socially sophisticated BDI agents. In *ICMAS*, pages 111–118. IEEE Computer Society, 2000.
- [10] J. Ferber, O. Gutknecht, and F. Michel. From Agents to Organizations: an Organizational View of Multi-Agent Systems. In *Agent-Oriented Software Engineering VI, LNCS 2935*, pages 214–230, 2004.
- [11] I. Foster, C. Kesselman, S. Tuecke, and D. Norrie. The Anatomy of the Grid: Enabling Scalable Virtual Organizations. *International J. Supercomputer Applications*, 15(3):200–222, 2001.
- [12] A. García-Camino, P. Noriega, and J. A. Rodríguez-Aguilar. Implementing norms in electronic institutions. In *EUMAS*, pages 482–483, 2005.
- [13] F. López y López, M. Luck, and M. d’Inverno. Constraining autonomy through norms. In *AAMAS*, pages 674–681. ACM, 2002.
- [14] F. López y López, M. Luck, and M. d’Inverno. A normative framework for agent-based systems. *Computational and Mathematical Organization Theory*, 12:227–250, 2006.
- [15] M. Luck, P. McBurney, O. Shehory, and S. Willmott. *Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing)*. AgentLink, 2005.
- [16] Y. Moses and M. Tennenholtz. Artificial social systems. *Computers and Artificial Intelligence*, 14(6), 1995.
- [17] B. Savarimuthu, S. Cranefield, M. Purvis, and M. Purvis. Role model based mechanism for norm emergence in artificial agent societies. In *COIN*, pages 1–12, 2007.
- [18] M. J. Sergot. Normative positions. In *Norms, Logics and Information Systems. New Studies in Deontic Logic and Computer Science*, pages 289–310. IOS Press, Amsterdam, 1998.
- [19] C. Sierra, J. Thangarajah, L. Padgham, and M. Winikoff. Designing Institutional Multi-Agent System. In *AOSE, LNCS 4405*, number 1-2, pages 84–103, 2006.
- [20] M. P. Singh. An ontology for commitments in multiagent systems. *Artif. Intell. Law*, 7(1):97–113, 1999.
- [21] J. Vázquez-Salceda, H. Aldewereld, and F. Dignum. Implementing norms in multiagent systems. In *MATES, LNCS 3187*, pages 313–327, 2004.

Distributed Barter–Based Directory Services

David CABANILLAS^a and Steven WILLMOTT^a and Ulises CORTÉS^a

^a *Universitat Politècnica de Catalunya, Software Department, E-08034 Barcelona*

Abstract.

The allocation of resources amongst a set of autonomous entities with contradicting interests is a recurring and relevant theme in distributed domains.

In this paper we present the design, implementation and evaluation of a *distributed directory service* based on a *bartering mechanism* in order to improve robustness in data retention. A *directory service* maps the names of network resources to their respective network addresses. A distributed directory service maintains this information not at a single node in a network, but across many nodes. The major challenge addressed in this paper is to build a workable system which not only responds to queries from users but A) ensures that directory items are never lost in the system and B) optimizes query response time with respect to different patterns of query arrival.

Keywords. Bartering, Directory Services

Introduction

The primary function of *directory services* is to repeatedly allocate a set of entries in accordance with clients demands at successive times. The basic model behind these services involves partial customer preferences over entries, and the directory service aims to satisfy these preferences as quickly as possible.

Dispersed pools of networked computing resources need *directory services* that store information about network resources. With the adoption of decentralization approaches in the distribution of administrative control, even though a common policy is adopted, no one individual entity is in control of the whole information. In such scenarios, all individuals work cooperatively following the same aim to respond to the queries delivered by the clients.

An autonomous and distributed barter–based implementation of the directory services combines [1] simplicity and distributed nature for barter. An additional benefit of bartering content is that its nature forces the nodes that store information to maintain entries in the system, making entries highly available and less likely to be lost due to failures.

The aim of the paper is to build a *distributed directory services* that:

- Manages the queries made by the clients using a team of cooperating and competing directory services
- Ensures that directory items are never lost in the system
- Optimizes query response time with respect to different patterns of query arrival

1. Related Work

A major drawback of existing large scale content distribution systems is the directory services, which generally consists on an index server and a tracker server. The index server (e.g., a web server) hosts all the metadata of shared content. In effect, such a directory services does not scale well as it cannot accommodate a large number of requests when the population of the system increases rapidly. In order to overcome this problem, many systems propose a decentralized service directory infrastructure ([2], [3]) such as Novell's NDS, Microsoft's Active Directory and others.

To improve the performance of large scale content systems, most of the work has been focused on keeping the cache information close to the client applications that access the directory information [4]. For example, to enhance web browsing, content distribution networks (CDNs) [5] move web content closer to clients by caching copies of web objects on thousands of servers worldwide. Additionally, to minimize client download times, such systems perform extensive network and server measurements, and use them to redirect clients to different servers over short time scales. CDNs include systems such as those provided by AKAMAI, Mirror Image, BitGravity, CacheFly, and LimeLight.

In our proposal, directory systems work by following a similar idea but applying a bartering mechanism ([6], [7]). The providers of entries want to have, or to have near, the content most requested by their clients, this proximity is achieved by exchanging entries with neighbors that follow the same strategy. Each self-interested provider/trader starts with some given initial bundle of entries. A new set of required entries, is build up from the clients queries. The providers discuss the proposal distribution among themselves taking the best choice for its clients (i.e. trying to get the most requested entries by its clients). If a provider/buyer decides that it can do better on its own, with its given initial entries, it makes a proposal of exchange that the other provider/seller should evaluate and this proposal only will be accepted if it is beneficial. When both parties accept the exchange, entries are transfered between them [8].

This paper is motivated by the need for autonomy in directory services [9] and provision architectures for mobile computing environments. In order to achieve these aims, a bartering strategy is used.

2. The DBBDS system

Distributed Barter-Based Directory Services (*DBBDS*) is an approach based on a set of interconnected peers called "Directory Nodes" (*DN*). Each *DN* only has partial knowledge of the network, no one has all the information/entries. A *DN* in *DBBDS* is part of a directory services team (see Figure 1). This team manages the queries requested by the clients of the directory service. This team is a community of cooperating and competing components. The obligations that any *DN* has as a member of the *DBBDS* are:

- To keep a set of entries
- To respond as fast as possible to the clients' queries

Each *DN* in the directory services has the following features:

- Each *DN* is autonomous and self-interested
- Links between *DN* are used to find a *DN* which can resolve a query

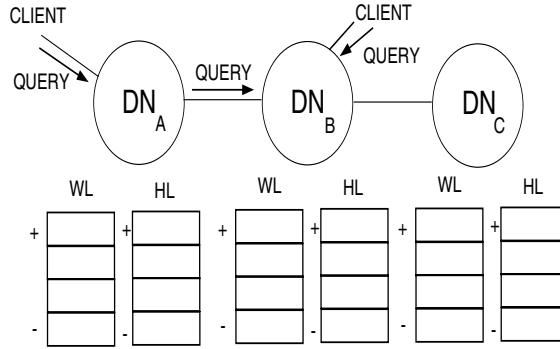


Figure 1. The *DN*s are linked shaping a directory services network. Each *DN* has a set of clients associated to it.

- Each *DN* takes local decisions. The information comes from requests from own clients and requests provided by neighbors
- Each *DN* keeps a list of entries and it is responsible for the storage of keys (i.e. similar to Chord). The only way to change an entry is by means of a bartering deal
- Each *DN* has limited resources such as storage capacity, information

The *DN*s have limited resources and they have a commitment to keep local entries as members of the *DBBDS*. Perhaps these entries are not useful for them at the current moment but these entries could be useful in the future, or necessary for other *DN*s. Under no circumstances should the *DN* remove an entry. As distributed cooperative directory services, the team of *DN*s should respond to any entry that can be requested by any client in the system at any time. The *DN*s keep the set of entries. If the storage capacity has reached the limit of entries that it can store, no more entries can be kept. The only way to exchange entries it is establishing an exchange with a neighbor (i.e. barter an entry for another entry).

The set of *DN*s are a collaborative network such as Internet e-mail. In e-mail there will only rarely be a direct connection between your network and the recipient's network, mail will make a number of stops at intermediate networks along the way. At each stop, another e-mail system temporarily stores the message while it figures out the best way to relay the message toward its ultimate destination. In *DBBDS* the *DN* aim is to respond as rapidly as possible to the clients' queries. For this reason, each *DN* desires to entries most requested by its own clients as near as possible at hand and at the same time not to be responsible of entries that are not interesting for its clients.

When a query can be directly responded to, the time to respond is equal to one tick. When this is not possible, the client's query is forwarded to the *DN*s neighbors increasing the response time. The further away the requested entry is, the more time it takes.

Queries that cannot be answered by the *DN* are re-sent to the *DN* neighbors. Once the client sends a query to the *DN* which it is related to, these *DN*s search in its have-list. If the requested entry is not in the have-list, the query is re-sent to the *DN*s neighbors until some *DN* has the entry or the life-time of the query expires (i.e. following

a flooding queries schema – the query is propagated to all neighbors within a certain radius). The information useful, and available to the DN_s are:

- Have-list (HL): The list of entries that the DN has
- Want-list (WL): The list of entries that the local and foreign clients want
 - * Local clients queries: Queries from clients directly connected to DN_s
 - * Foreign clients queries: Queries come from clients of others DN_s

Each one of these lists is composed of two components:

- Node directory entry: Contains the address for each remote workstation
- Request rate (RR): This component defines the order in the list

Time window (TW): The time window is employed to balance new information against past experience. A request is limited to a specified time window beginning with time t_1 , and extending up to time t_2 (i.e. window time interval $[t_1, t_2]$). Following this approach, the oldest requests will be removed from the WL replaced by new requests.

Query distribution (QD): The query distributions that the population follows:

- An uniform query distribution: All the entries have equal probability for getting requested.
- A Zipf query distribution: In Zipf-like distribution, the number of queries to the i 'th most popular object is proportional to $i^{-\alpha}$, where α is the parameter of the distribution. The query distribution has a heavier tail for smaller values of the parameter α ([10], [11]).

A Zipf distribution with parameter 0 corresponds to a uniform distribution and with a value α equals to the unity follows a Zipf distribution.

Content Distribution (CD): The volume and type of content each DN carries.

Request Generation Rate (RGR): Clients in a $DBBDS$ issue queries to search for entries that match their interests. A client without an entity will generate a search request for the entity at certain rate depending on the preference for the entity. Each client i is assigned a query generation rate q_i , which is the number of queries that client i generates per unit time.[12]

Pressure of foreign queries (PFQ): This parameter allows the importance/significance of the external queries with respect to the local queries to be set up

- $\lambda_f = 0$ the queries from foreign clients/ DN have the same importance than the queries from local clients.
- $\lambda_f = 1$ the queries from foreign clients/ DN have less importance than the queries from local clients.

Topology (T): The topology of the network establishes the links from peers to a number of peers.

Time to Live (TTL): Before resending a query the TTL field is increased if the new value is higher than a certain threshold the query is not resent. Otherwise, the query is resend to the DN_s neighbors.

Updating WL and HL : The process to update the order in WL and HL follows the algorithms proposed in algorithm 1. In the algorithm Q_L is a query from a local client. For foreign queries the algorithm is the same than for the local query except for the first conditional that it does not appear and changing Q_L by Q_F that is the foreign query. The

local queries update the utility/satisfaction in the WL and HL of the entry associated to the query. For foreign queries only the WL is updated because the HL is restricted for local queries.

Algorithm 1 Local Query

```

if  $Q_L \in HL$  then
  to update the rate request of  $Q_L$  in  $HL$ 
   $HL \Leftarrow Q_L$ 
end if
if  $Q_L \in WL$  then
  to update the rate request of  $Q_L$  in  $WL$ 
end if
 $WL \Leftarrow Q_L$ 

```

3. Implementation Overview

In order to analyze our model, we conducted simulation experiments to judge what is the performance of query response time and how content is distributed and re-allocated in the system. Simulations were performed to assess the effectiveness of the service directory infrastructure. We assume that the agents themselves are reasonably long-lived and static.

A simulation starts by placing the m distinct entries randomly into the DNs network. Then the clients start to generate queries according to a Uniform/Zipf-like process with average generating rate at a queries per tick to its DNs . These queries are analyzed by the DNs . In case that the query is one of the entries in the HL , the query is responded to and the entry updated (i.e. increasing its value). If DN does not have the entry, the DN resend the query to its neighbors and it updates the WL . With the information provide from HL and WL , DNs offer to its neighbors the entries that they want and it does not want. The only way to get valuables entries is offering valuables entries to the neighbors. Therefore, the process to update the HL and WL allows to DNs distinguish between devaluated and not devaluated entries in order to establish beneficial exchanges. The simulation finishes when all the clients queries are processed or time finish is reached.

4. Experiments

The paper provides experimental results for the following parameters:

- TW : 8.
- QD : Random distribution and perfect Zipf distribution. In this latter case, the clients are interested in a set of entries and this set is not in common with the rest of clients of others DNs (i.e. disjoint sets where each DN has a set of entries with no element in commons with another DN and the clients only wants entries of the this set following a Zipf query distribution).

- *CD*: In this work we consider the *DN*s homogeneous in terms of type and volume of data each *DN*s carries the same quantity of entries, the simulated scenarios have 1,000 entries distributed in 100 entries per *DN*.
- *RGR*: Each *DN* only has an unique client. And each client only wants an entry per unit of time following the *QD*.
- *PFQ*: The clients of each *DN* are only worried in an exclusive set of entries in the proposed simulations this parameter is not relevant. However, when clients of different *DN*s want the same entry, the pressure of foreign queries parameter will affect the performance in the system.
- *T*: 100 *DN*s following an Erdos–Renyi structure (see Figure 1). The number of unreachable pairs is equal to 0. The average distance among reachable pairs is 3.6 and the most distant between vertices is 8.
- *TTL*: *TTL* with values from 0 to 5.

And the performance parameters studied are:

- Response Time: The response time is defined as the number of hops between the source and the destination.
- Percentage Success Rate: The percentage of requests that are responded to.
- Quantity of exchanges: The quantity of exchanges (i.e. one entry by other entry) that are made in the system.

The two first parameters are related to service quality, the third is related to the performance of the exchange strategy.

4.1. Random and perfect Zipf query distributions

In this section, the worst and the best scenario are compared for an exchange-based approach. If the preferences of the clients follow a random distribution, the *DN* can not place the preferences in a stable order. The bartering mechanism is a very attractive form of exchange, but each party needs to know the devalued and appreciated properties of the entries and this knowledge should be stable enough to be applied to the exchange process by the *DN*. On the other hand, in a perfect Zipf distribution, with the passing of time, *DN* knows what are the needs of their clients, keeping the most valuable entries and using the rest to barter. Also, in a perfect distribution there is no competition amongst *DN*s.

Figure 2 shows the performance of the system when the clients follow a random query distribution. The exchange policy, in this case, neither does not get positive results in cases where $\lambda_I = 0$ or $\lambda_I = 1$. Also, in both cases the quantity of exchanges is significant. On the contrary, in figure 3 using a policy exchange improves the performance. Concretely, the policy exchange can shorten the query response time by about 29 % when $TTL = 5$, about 14 % when $TTL = 4$. At the same time, the query success rate improves by 16 %, 27 % with TTL equals to 4 and 3 with λ_I equals to 1. For lower TTL s values the exchange policy does not improve significantly due to the inability to make many changes. If the quantity of neighbors *DN*s that knows the needs of a *DN* is reduced. Comparing the pressure of foreign queries, with $\lambda_I = 1$ the quantity exchange decreases due to the *DN*s giving more priority to the queries from own clients than queries from foreign clients but the performance is better. This strategy tries to exploit spatial locality.

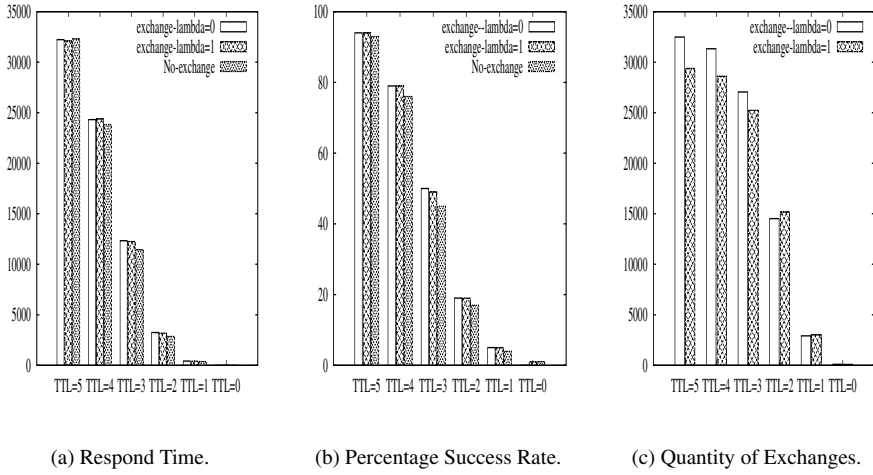


Figure 2. Query random distribution.

Comparing the values in figures 2 and 3 it can be seen that the first scenario corresponds to the worst scenario, and the second one to the best scenario for the barter-based approach. When the clients follow a query random distribution both parameters, response time, and percentage success response with or without exchange policy, are similar. However, when the clients follow a perfect Zipf query distribution, the response is reduced and the percentage success response is increased. The performance parameter that measures the quantity of exchanges shows that with a random query distribution the quantity of exchanges is much greater than with a perfect Zipf query distribution. The reason for this poor performance, and the large number of exchanges, is due to the fact that in a random query distribution the *DNs* have an unstable list of what the clients wants. This fact implies that they are trying to get many different entries and usually they have not these requested entries. On the other hand, when the clients follow a perfect Zipf query distribution, the *DNs* only keep a stable list of entries that the clients want and the *DNs* are not in conflict with others *DNs*. These two factors facilitate the improvement of the performance.

5. Conclusions and Future Work

The following properties were obtained from the previous results:

- The topology affects the information that *DNs* should evaluate in the exchange process.
- The *TTL* parameter limits the propagation of the query. In our case however, it also limits the propagation of the updating of the *WL* and the opportunities to establish exchanges.
- Increasing the *TW*, the *WL* holds the clients' requests longer. This information increases the opportunities to make exchanges.

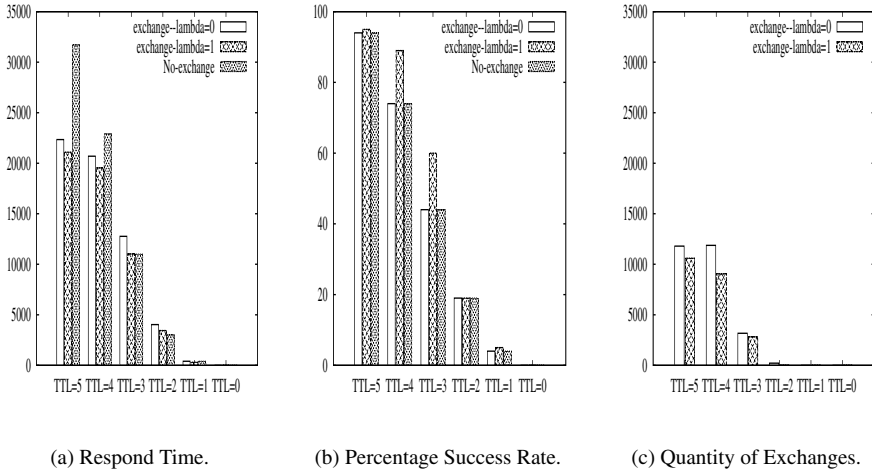


Figure 3. Query Zipf distribution.

- Due to all the clients of a same *DN* follow the same Zipf-local query distribution the pressure in the *DNs* (i.e. in the *DNs'* *WL*) increases, thus increasing the response queries and diminishing the quantity of messages. Turning perfect query distribution to non perfect one that will have a negative effect on the performance.

Barter-based systems bring several advantages. Firstly, they preserves the autonomy of individual participants. Each participant makes local decisions about who to trade with, how many resources to contribute to the community, how many trades to try to make, and so on. Secondly, the symmetric nature of trading ensures fairness and discourages free-loading (i.e. bartering is an incentive scheme by nature). In order to acquire resources from the community, each participant must contribute its own resources in trade. Moreover, participants that contribute more resources receive more benefit in return, because they can make more trades. Thirdly, the system is robust in the face of failure.

The advent of powerful computing facilities in the participants has enabled two important paradigm shifts over the last decade. The first shift is to move away from categorizing entities according to the traditional client-server model, and the second is the progressive adoption of decentralized overlay systems. Both paradigm shifts dramatically change the way in which communication systems are designed and build; and both are pertinent to the realization of truly autonomic communication systems. The adoption of further decentralization, which in part is expedited by the desire to utilize the improved capabilities of end hosts, allows the distribution of functionalities across a subset or the whole of the participating end hosts, providing the advantage of robustness by removing single-point failures in the system. Furthermore, the resources, and thus the cost, required to provide the functionality can be distributed to all participants. Finally, decentralization results in the distribution of administrative control so even though a common policy is adopted, no one individual participant is in control of the whole system. Therefore, the major challenge in the implementation of a directory decentralized system is to build a system that works correctly without the need for a central coordination unit.

The distribution of a set of entries amongst a set of distributed and autonomous agents, with varying preferences, is a complex combinatorial problem. Bartering can be used as a form to resolve this problem. In barter exchange each party uses a devaluated currency, in some cases one that would otherwise be wasted. The unused entries in your basement can be converted into something you need or want. Likewise, the party with whom you are swapping is giving something that has a greater value to you than it has for them.

Future research includes the study of interest based communities, users that in one cluster share a subset of common entries and are likely to be of interest to other entries popular in the cluster. The transitivity property may be used for enabling private information between users, in order to suggest entries that are potentially of interest for members of the same cluster [13]. Other aspect to study is the tolerance of faults [14].

6. Acknowledgements

The research described in this paper is partly supported by the European Commission Framework 6 funded project CONTRACT (INFSO-IST-034418). The opinions expressed herein are those of the named authors only and should not be taken as necessarily representative of the opinion of the European Commission or CONTRACT project partners.

References

- [1] Tim Hsin-Ting Hu and Aruna Seneviratne: Autonomic Peer-to-Peer Service Directory IEICE - Trans. Inf. Syst. (2005)
- [2] Russ Cox and Athicha Muthitacharoen and Robert Morris: Serving DNS Using a Peer-to-Peer Lookup Service IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems. (2002)
- [3] Cezary Dubnicki and Cristian Ungureanu and Wojciech Kilian: FPN: A Distributed Hash Table for Commercial Applications HPDC '04: Proceedings of the 13th IEEE International Symposium on High Performance Distributed Computing. (2004)
- [4] Sophie Cluet and Olga Kapitskaia and Divesh Srivastava: Using LDAP Directory Caches. (1999)
- [5] Ao-Jan Su and David R. Choffnes and Aleksandar Kuzmanovic and Fabián E. Bustamante, Drafting behind Akamai (travelocity-based detouring). (2006)
- [6] C. Krauss: To Weather Recession, Argentines Revert to Barter. (2001)
- [7] David J. Abraham and Avrim Blum and Tuomas Sandholm Clearing algorithms for barter exchange markets: enabling nationwide kidney exchanges. (2007)
- [8] Jaeyeon Jung and Emil Sit and Hari Balakrishnan and Robert Morris: DNS performance and the effectiveness of caching IEEE/ACM Trans. Netw. (2002)
- [9] Zhang Qiang and Zhao Zheng and Yantai Shu: P2PDNS: A Free Domain Name System Based on P2P Philosophy Proceedings of the Canadian Conference on Electrical and Computer Engineering. (2006)
- [10] Lee Breslau and Pei Cao and Li Fan and Graham Phillips and Scott Shenker: Web Caching and Zipf-like Distributions: Evidence and Implications, INFOCOM. (1999)
- [11] Venugopalan Ramasubramanian and Emin Güzın Sirer: Beehive: O(1) Lookup Performance for Power-Law Query Distributions in Peer-to-Peer Overlays
- [12] Vern Paxson and Sally Floyd: Wide area traffic: the failure of Poisson modeling IEEE/ACM Transactions on Networking. (1995)
- [13] B. Logan and G. Theodoropolous: The Distributed Simulation of Multi-Agent Systems Proceedings of the IEEE, 89(2):174-185. (2001)
- [14] M. Frans Kaashoek and Andrew S. Tanenbaum and Kees Verstoep: Using Group Communication to Implement a Fault-Tolerant Directory Service. (1993)

Trading Paper Clips – An Analysis of “Trading Up” in Artificial Societies without Altruists

David CABANILLAS^a and Steven WILLMOTT^a and Ulises CORTÉS^a

^a *Universitat Politècnica de Catalunya, Software Department, E-08034 Barcelona*

Abstract. The remarkable recent story of Kyle MacDonald who, by means of a sequence of bartering exchanges between July 2005 and July 2006, managed to trade a small red paperclip for a full sized house in the town of Kipling Saskatchewan has inspired interesting debates about the nature of such feats and questions as to how they might come about. While bartering is arguably the world’s oldest form of trade, there are still cases such as this which surprise us. Although there are many factors to consider in Kyle’s achievement, his feat raises basic questions about the nature of the trades made and to what extent they are repeatable.

In this paper we provide an intuitive model for the type of trading environment experienced Kyle and study its consequences in order to analyze in particular whether such trading phenomena require *altruistic agents* to be present in the environment and extending the experience with multiple goal seeking agents.

Keywords. Bartering, One Red Paperclip, Electronic Market, Arbitrage

Introduction

Although the motivation for making trades amongst the participants in Kyle’s story¹ is unknown, some of them may have been motivated by *altruism*. In this case, altruism can be defined as a willingness to accept something of lower value in order to help Kyle on his way or to obtain other peripheral secondary in-direct benefits (such as a desire to participate in an interesting experiment). However, it is likely that the majority of participants were probably making trades in which they at least sought significant value (if not full value) – Kyle was deliberately seeking out potential exchange partners who valued his current item the most. Further, while the motivations of the original participants are unknown, a key question in such scenarios is – “Would such a general mechanism work if there were no altruists at all?”. Scenarios where self-interested agents barter/exchange resources in order to increase their individual welfare are ubiquitous ([1], [2]) and hence it is of interest to understand whether such trading patterns could arise.

This paper develops a simple agent population model based on active/goal-driven and passive agents with ranges of personal value distributions for the items they own

¹One Red Paperclip: Or How an Ordinary Man Achieved His Dream with the Help of a Simple Office Supply.

and uses a simple trading mechanism to show that scenarios such as Kyle’s story are indeed possible for goal-driven agents without relying on altruistic behavior. The work characterizes the conditions necessary for this to occur and goes on to study the emerging dynamics as an increasing number of agents become active.

1. Related Work

The increasing popularity of P2P networks and other such forms of distribution networks, has re-invented the bartering model and made it relevant to the modern technological world (see [3], [4], [5], [6]). Examples are present in many different areas such as file sharing [7], query forwarding [8], routing [9], knowledge diffusion [10], storage-sharing systems [11], and WIFI hotspot sharing [12] applied in commercial platforms like Linspot², Netshare³ or Fon⁴. Barter has also been used in B2B commerce with many others examples such as BizXchange, ITEX, BarterCard and Continental Trade Exchange. Therefore, many hopes are riding on barter mechanisms in the Internet Age.

The One Red Paperclip scenario is a classic example of bartering and arbitrage ([13], [14]) – where value is extracted by playing on the asymmetries of users valuations. Betting exchanges have many similarities to the Kyle’s experiment. Betfair⁵, Betdaq⁶ and other similar betting exchanges have huge turn over now and many billions of pounds are matched each month on these markets. In betting exchanges an arbitrageur exploits existing price discrepancies when bookmakers’ prices differ enough that they allow to back all outcomes and still make a profit. In paperclip exchanges Kyle exploits personal values discrepancies. Both Kyle and sports betting take advantage from the personal valuation differential between agents in large-scale markets. But there are still barriers which stop everyone from being successful in both scenarios. Both scenarios require capital, time, organization and energy to make profits.

2. Formal definition

The model developed for the scenario is relatively simplistic, but captures the main elements of Kyle’s trading environment. The model consists of the following components:

- A population of agents in which each agent plays one of two roles:
 - * *Goal-driven agents (GDA)*: These agents try to reach an item with a value that seems infinite to them and is also very high on the general market value ranking. The initial property of this type of agent is considered low in the general market ranking. Looking for rich/beneficial trading encounters in order to move upwards in market value.
 - * *Passive agents (PA)*: These agents have an item and do not seek any new concrete item, however they know a good deal when they see one. In case that

²Linspot by Biontrix <http://www.linspot.com>

³Netshare by Speakeasy Inc. <http://www.speakeasy.net/netshare>

⁴Fon in <http://www.fon.com>

⁵Betfair in <http://www.betfair.com>

⁶Betdaq in <http://www.betdaq.com>

a *GDA* tries to trade with a *PA*, the *PA* only accepts trade if it is indeed beneficial.

- A list of items: An item is anything private goods like food, clothing, toys, furniture, cars etc. This list follows a strict order in function of a general *market value* (*MV*). This value is equal for any agent in the market.
- Each agent has a *personal value* (*PV*) for each item in the market (and hence for each item they own). This *PV* differs for each agent in the market with a statistical deviation which may be positive or negative – in other words an agent may value certain items at above or below general *MV*. $MV_i(g_j)$ and $PV_i(g_j)$ represent the *MV* and *PV* respectively of the *agent*_{*i*} with respect to the item *g_j*.
- Each agent is connected to the rest of the members in the market.
- A set of ranges: A range contains multiple items with the same *MV* and a range of possible *PV* restricted to two values $[-\sigma, +\sigma]$ related to this *MV*. Without this clustering the cost to finding all possible ways can be too expensive
- *The exchange strategy*: An exchange between two agents *GDA* and *PA* is accepted iff there exist two items g_i, g_j , where $j=i+1$ that are in neighboring ranges such as:

$$\{PV_{PA}(g_i) > PV_{PA}(g_j) \text{ and } MV_{GDA}(g_j) > MV_{GDA}(g_i)\}. \quad (1)$$

3. Experiments

The paper provides experimental results for a number of cases based on the following general parameters and for the cases of a single *GDA* and of multiple *GDA*s.

- Initially, items are randomly assigned to agents. One item per agent.
- The market is composed of five thousand agents.
- The number of ranges is fifty. Each range is composed of one hundred items. In range₁ there are the items with smaller value and in the range₅₀ the items with the higher value.
- *GDA*s know where the rest of the agents are.
- Items have a unique *MV* but each agent has its *PV* for each item in the market.
- Trades are conducted by means of bartering. An exchange always is between a *GDA* with an item from range_{*x*} and a *PA* with an item from range_{*x*+1} (see Figure 1).
- *GDA*s only take local decisions.
- *GDA*s only trade when the interchange is immediately beneficial according to general *MV*. The *PA*s only trade when the interchange is immediately beneficial according to its *PV*.
- The *PV* of the items follows a $\sim \mathcal{N}(\mu, \sigma)$. Then, $\mu_i - \mu_{i+1}$ represents the distance between ranges or between cluster of items with the same *MV* and σ represents the variation of *PV*.
- In each of our graphs, each data point is an average of ten simulations, and we provide statistical significance results to support our main conclusions.
- A *blocking situation* is when a *GDA* wants some item from a range but no one in this range wants to trade. This is because the agents in the ranges are *GDA*s or the item offered is not good enough for the *PA*s in the next range.

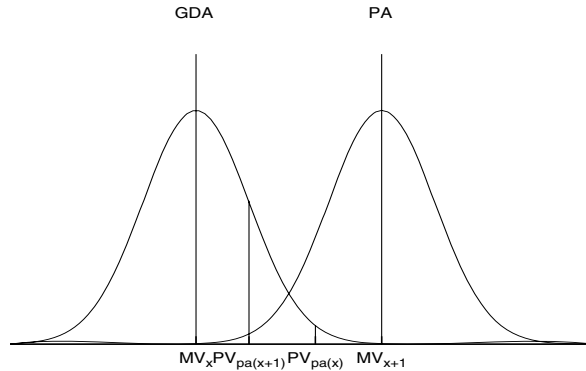


Figure 1. The *GDA* is increasing its *MV* and the *PA* is decreasing its *MV* but it is increasing its *PV*.

- A *steady state* is achieved when the *GDA*s reach the desired item or when *GDA*s stay in a blocking situation or when the simulation deadline pass fifty steps.

The parameters analyzed in the environment are:

- **Quantity of items and agents:** The quantity of agents and items in the market have a great impact on the performance of the market. Finding a profitable exchanges for buyer–seller (i.e. double coincidence of wants) depends on how many members are shaping the market.
- **Range of values:** One way to analyze the quantity and distribution of items in the market is in fixed ranges of value – each representing different levels of value and containing multiple items in one range. In a range with few items the distance between the *MV* increases and when the quantity of items increases the distance is reduced.
- **The distribution of *MV* and *PV*:** Two different probabilities are related with respect to this model based on ranges.
 1. Inter–range: The probability of reaching the desired object starting with the chain of trades from the worst item (i.e. from range₁ to range_Z). This is a binomial random variable, a random variable that counts the number of successes in a sequence of independent Bernoulli trials with fixed probability of success. In our case, the probability of passing to the next range (i.e. of having a successful jump or not).
 2. Intra–range: This corresponds to the probability that a *GDA* finds a *PA* to interchange the items with the next range.
- **Single and multiple *GDA*s:** The most basic form of the systems to be explored is that in which there is only a single *GDA* looking for a desired item which has the highest value in the market (i.e. from a paperclip to a house). Once proven that an isolated *GDA* can reach an item from the last range under some configurations, the next step is to balance the quantity of *PA*s and *GDA*s.

Figure 2 covers the case with a single *GDA*. Combining the parameters of the environment in order to check the probabilities to success that the *GDA* has.

Figure 2 a) shows the effect of the quantity of items per range. The only two parameters modified are: the quantity of items per range and the distance between ranges. The

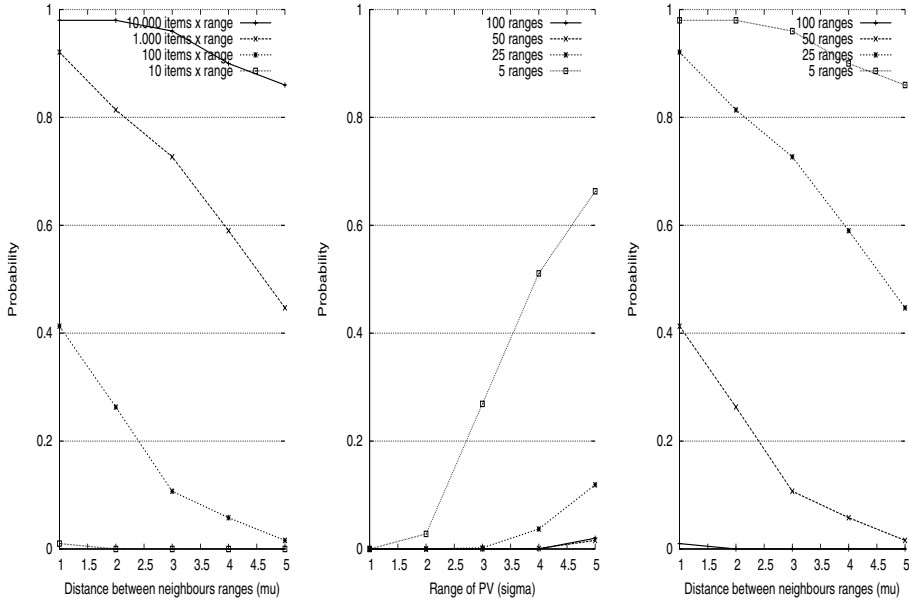


Figure 2. Results related to the parameters in the simulator a) items per range b) variations of PV and c) distance between ranges.

rest of the parameters are fixed. Simulations are related to the case where the distance between the lower range and the higher range is equal to fifty (i.e. fifty hops are necessary to transform a paperclip into a house) and the range of PV is equal to five. Figure 2 shows that as there are more items per range there are more probabilities that the GDA will reach the last range and thus more access to the most valuable items. Also the figure shows that in some configurations (for example – with few items per range as 10 items x range), the probability of reaching the last range is near to zero. In other configurations, for example with 1,000 items x range, a range of PV equal to 5 and a distance between ranges equals to 2, this probability is high but not 1 – in this case 0.82. Figure 2 b) shows different variations of PV from 1 to 5 and the quantity of ranges. The rest of parameters are: quantity of items per range is 1,000 and the distance between ranges is equal to 5. Increasing the value of PV the probability of reaching the last range increases. Finally, figure 2 c) shows the effect of the distance between ranges combined with the quantity of ranges. The probability of reaching the last range decreases as distances between ranges increase or the quantity of ranges increases. The variation of PV is fixed to 5 and the quantity of items per range is equal to 1,000. As the number of ranges to cross over is lower, it is easier to reach the last range. It could be noted that as the distance decreases between ranges it becomes easier to get an item from the last range.

Once proven that an isolated GDA can reach an item from the highest range under some configurations (i.e. showing that Kyle’s feat is possible), the next step is to balance the quantity of PAs and $GDAs$, to check the behavior of the market with other distribution populations. Therefore, the strategy is to increase the percentage of $GDAs$ in the market in order to reveal the dynamics that appears in front of the variation of populations.

The set of experiments uses configurations with a percentage ranged from 0, 0.02, 2, 10, 20, 30, 40, 50, 60, 70, 80, 90 to 100 % *GDA*s. Other parameters are set as follows: the variation of *PV* is equal to five, the distance is equal to five (i.e. difference between two consecutive *MV*). These parameters are chosen from the previous section because they form a fruitful environment where trades with one *GDA* can be made. These results are presented in Figure 3 where the quantity of crossed ranges or jumps is shown with respect to the percentage of *GDA*s. The solid line is related to the maximum sum of jumps. This value captures starting from a random distribution of the *GDA*s in the different ranges, how many crossed ranges should be crossed to become this initial situation in a situation where all the *GDA*s have the best available items. On the other hand, the dotted line is related to the sum jumps that were obtained by simulations.

Focusing on this later value, the figure shows that when the percentage is reduced (i.e. less than 2 %) the value of jumps in our simulator and the maximum value expected is equal. The best results with respect to the quantity of crossed jumps are achieved when the balance of *GDA*s is around 10 %. The reason is because many *GDA*s are making jumps but not enough to decrease the opportunities to make exchanges from the rest of *GDA*s in the market. Under other configurations this property is not applicable. As the quantity of *GDA*s increases in the market the sum of jumps go down slightly. At first glance, more *GDA*s in the market should implies that more jumps could be done, the problem is that the opportunities of jumps decreases, ending up with the opposite of the expected value.

The results are shown in Figure 3 and demonstrate that from 0 to 10 % of *GDA*s the mean range value improves but once the percentage of *GDA*s exceed 10 %, the mean range value decreases. Therefore, as the number of *GDA*s increases, it is more difficult to make trades between agents. The reasons include:

- As the number of *GDA*s increases, there is a gradual reduction in *PA* agents and hence opportunities for trade in general decrease.
- Once a *PA* makes a trade the following events occur:
 - * The *PA* increases the *PV* of its current object.
 - * The *PA* moves downwards by one range.

Each of these factors affects the dynamics of the trading environment, gradually affecting the opportunity to trade for all agents. This leads us to the conclusion that a small number of goal-seeking agents can gain increasing benefit, but if the number increases utility is reduced/throttled. Also, *GDA*s that start with better items have more opportunities to get an item from the last range because they should do less exchanges and they are, in many times, the firsts to establish exchanges with *PA*s in upper ranges. Further dynamics are seen in cases of agents using backtracking or being able to repair items they own to increase their value.

4. Conclusions and Future Work

In the modeled market, there are sequences of trades that turn an item from range_{*x*} into an item of the highest range. However, a number of conditions need to be met in order for *GDA*s to be able to make these trades and in particular the following parameters are of relevance:

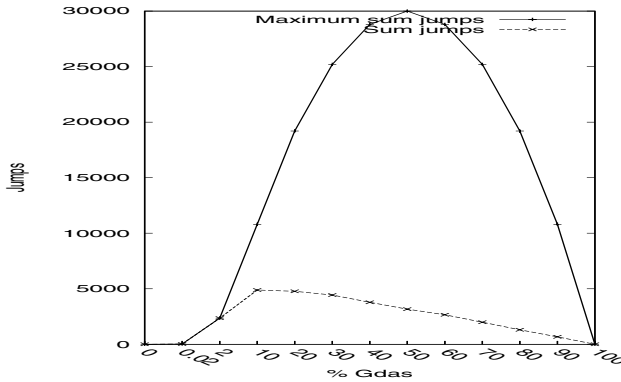


Figure 3. The mean range value decreases as quantity of *GDAs* increases.

- **The distance between *MVs*:** As this distance increases it is more difficult to change an item – with increasing gaps between valuations.
- **The variance of *PV*:** The greater the variance in the *PV*, the greater the probability that a *PAs* will be interested in to interchanging items – since some outliers will have very high valuations.
- **The quantity of items per range:** A market where the quantity of items is great will increase the possible chains to reach the target item.
- **The quantity of ranges:** The fewer the ranges the easier it is for *GDAs* to have access to the last range where the desired item resides (in fact this parameter varies with the distance between ranges).
- **The quantity of *GDAs* in the market:** The more *GDAs* there are, the more competition there is since many *GDA* may be trying to get the best items in the market. On the other hand, the quantity of *PAs* increases the opportunities for the *GDAs* to trade up.

Returning to Kyle’s story and reviewing the results in Figure 2 we can see that Kyle’s feat is possible. Considering that this took place in a network where the quantity of possible trading partners was extremely large (a significant proportion of the population with Internet access in the United States), the probability that by means of twenty trades Kyle can reach his objective was relatively high. However when there are many *GDAs* in the market, it becomes clear that it is not necessarily possible to repeat the Kyle’s behavior over and over again, Figure 3.

Kyle’s environment differs from our environment in two main points:

1. The quantity of agents and items tends to infinity. Also, the market is composed of one *GDA* and the rest are *PA*. But this is only one instance of the proportions of agents that a market can have. For example on www.eBay.com there is a red paperclip on sale for \$1 but nobody offers even this \$1.
2. Some agents accept trades that are not beneficial to them. At least not beneficial with respect to the established/general economical rules (i.e. the agent gives more value than that it receives in exchange). Merely evaluating the value of the item in the exchanges way lead us to assume that a seemingly altruistic exchange has occurred. However, as we should always bear in mind that the goal of the *GDA* is a final objective, other factors need to be taken into account when evaluating

the exchange. These may include publicity, excitement and so on compensating the seller and turning an altruistic exchange into a beneficial exchange.

Kyle’s and other similar experiences show alternative economic visions to normal electronic transaction which are anonymous and money oriented, by relying on personal encounters which are mediated by useful trades for both parts of the negotiation. This basic trading approach opens news opportunities in large scale exchange environments.

Future research includes the exploration of other modeling choices, such as: Non-linear value ranges, opportunistic *GDA*s, and factoring in the cost of exploration/discovery of tradable items.

5. Acknowledgements

The research described in this paper is partly supported by the European Commission Framework 6 funded project CONTRACT (INFSO-IST-034418). The opinions expressed herein are those of the named authors only and should not be taken as necessarily representative of the opinion of the European Commission or CONTRACT project partners.

References

- [1] C. Krauss: To Weather Recession, Argentines Revert to Barter. (2001)
- [2] Sujay R Sanghavi: How to Exchange Items. (2004)
- [3] Chiranjeev Buragohain and Divyakant Agrawal and Subhash Suri: A Game Theoretic Framework for Incentives in P2P Systems Proceedings of the 3rd International Conference on Peer-to-Peer Computing. (2003)
- [4] Philippe Golle and Kevin Leyton-Brown and Ilya Mironov: Incentives for sharing in peer-to-peer networks Proceedings of the 3rd ACM conference on Electronic Commerce. (2001)
- [5] K. Ranganathan and M. Ripeanu and A. Sarin and I. Foster Incentive mechanisms for large collaborative resource sharing Proceedings of the 2004 IEEE International Symposium on Cluster Computing and the Grid. (2004)
- [6] David J. Abraham and Avrim Blum and Tuomas Sandholm: Clearing algorithms for barter exchange markets: enabling nationwide kidney exchanges Proceedings of the 8th ACM conference on Electronic commerce. (2007)
- [7] Kostas G. Anagnostakis and Michael B. Greenwald: Exchange-Based Incentive Mechanisms for Peer-to-Peer File Sharing Proceedings of the 24th International Conference on Distributed Computing Systems. (2004)
- [8] Levente Buttyán and Jean-Pierre Hubaux: Stimulating cooperation in self-organizing mobile ad hoc networks. (2003)
- [9] Alberto Blanc and Yi-Kai Liu and Amin Vahdat: Designing incentives for peer-to-peer routing INFOCOM. (2005)
- [10] Cowan, Robin and Jonard, Nicolas: Network structure and the diffusion of knowledge Journal of Economic Dynamics and Control. (2004)
- [11] Brian F. Cooper and Hector Garcia-Molina: Peer-to-peer data trading to preserve information. (2002)
- [12] Elias C. Efstathiou and George C. Polyzos: A peer-to-peer approach to wireless LAN roaming Proceedings of the 1st ACM international workshop on Wireless mobile applications and services on WLAN hotspots. (2003)
- [13] John K. Debenham: Identifying Arbitrage Opportunities in E-markets. Proceedings of the Third International Conference on E-Commerce and Web Technologies Springer-Verlag. (2002)
- [14] Spear, Stephen E.: Comment on “Markets Come to Bits” Journal of Economic Behavior & Organization. (2007)

Validation and Experimentation of a Tourism Recommender Agent based on a Graded BDI Model

Ana CASALI ^{a,1}, Lluís GODO ^b and Carles SIERRA ^b

^a *Depto. de Sistemas e Informática, FCEIA - UNR, Rosario, Argentina*

^b *Artificial Intelligence Research Institute (IIIA), CSIC, Barcelona, Spain*

Abstract. In this paper, a validation and an experimentation of the use of graded BDI agents is reported. This agent model has been proposed to specify agents capable to deal with the environment uncertainty and with graded attitudes in an efficient way. As a case study we focus on a Tourism Recommender Agent specified using this agent model. The experimentation on the case study aims at proving that this agent model is useful to develop concrete agents showing different and rich behaviours. We also show that the results obtained by these particular recommender agents using graded attitudes improve those achieved by agents using non-graded attitudes.

Keywords. graded BDI agent model, experimentation, recommender system, tourism.

1. Introduction

In the last years, an increasing number of theories and architectures have been proposed to provide multiagent systems a formal support, among them the so-called BDI architecture [7]. This model has evolved over time and has been applied, to some extent, in several of the most significant multiagent applications developed up to now. With the aim of making the BDI architecture more expressive and flexible, in [1] a general model for Graded BDI Agents (g-BDI) has been proposed, specifying an architecture able to deal with the environment uncertainty and with graded mental attitudes. As a case study, a Tourism Recommender multiagent system has been designed and implemented where its main agent, the Travel Assistant Agent (*T-Agent*), has been modelled using the graded BDI model [2,3]. Actually, recommender systems [5] is an increasing area of interest within the Artificial Intelligence community where Agent technology becomes very valuable as it eases the expression of those different characteristics we expect from these systems (e.g. user profile oriented, able to aggregate relationships from heterogeneous sources and data, open and scalable) [6]. We have developed an agent-based tourism recommender system that has the goal of recommending the best tourist packages provided by different tourist operators, according to user preferences and restrictions. Particularly, we have implemented a case study on Argentinian destinations.

¹Corresponding Author: Ana Casali, Depto. de Sistemas e Informática, Facultad de Cs. Exactas, Ingeniería y Agrimensura, UNR, Av Pellegrini 250, 2000 Rosario, Argentine; E-mail: acasali@fceia.unr.edu.ar.

In this work we report on the validation process of the recommender system as well as on the experimentation we have performed with the aim of proving different properties of the g-BDI model of agents. Namely, we have performed a sensitivity analysis to show how the g-BDI agent model can be tuned to have different behaviours by modifying some of its component elements. Also, we have done some experiments in order to compare the performance of recommender agents using the g-BDI model with respect to agents without graded attitudes. This paper is structured as follows. In Section 2 the g-BDI model of agent is succinctly described. Then, in Section 3 we present the relevant characteristics of the Tourism Recommender implementation. In Section 4, we describe the validation process of the *T-Agent* designed using the g-BDI model and implemented in a multithreaded version of Prolog. Finally, the results of the above mentioned experiments are reported in Section 5. We conclude in Section 6 with some final remarks.

2. Graded BDI agent model

The graded BDI model of agent (g-BDI) allows us to specify agent architectures able to deal with the environment uncertainty and with graded mental attitudes. In this sense, belief degrees represent to what extent the agent believes a formula is true. Degrees of positive or negative desire allow the agent to set different levels of preference or rejection respectively. Intention degrees give also a preference measure but, in this case, modelling the cost/benefit trade off of reaching an agent's goal. Thus, agents showing different kinds of behaviour can be modelled on the basis of the representation and interaction of these three attitudes.

The specification of the g-BDI agent model is based on Multi-context systems (MCS) [4] allowing different formal (logic) components to be defined and interrelated. A particular MCS specification contains two basic components: contexts and bridge rules, which channel the propagation of consequences among theories. Thus, a MCS is defined as a group of interconnected units or contexts $\langle \{C_i\}_{i \in I}, \Delta_{br} \rangle$. Each context C_i is defined by a tuple $C_i = \langle L_i, A_i, \Delta_i \rangle$ where L_i , A_i and Δ_i are the language, axioms, and inference rules of the context respectively. Δ_{br} is a set of bridge (inference) rules, that is, rules of inference with premises and conclusions in possibly different contexts. When a theory $T_i \subseteq L_i$ is associated with each unit, the specification of a particular MCS is complete. In the g-BDI agent model, we have *mental* contexts to represent beliefs (BC), desires (DC) and intentions (IC). We also consider two *functional* contexts: for Planning (PC) and Communication (CC) and a set of bridge rules (Δ_{br}). Thus, the g-BDI agent model is defined as a MCS of the form $A_g = (\{BC, DC, IC, PC, CC\}, \Delta_{br})$. The overall behaviour of the system will depend on the logic representation of each intentional notion in the different contexts and the bridge rules. The left hand side of Figure 1 illustrates the g-BDI agent model proposed with the different contexts and some of the bridge rules relating them. As for example, we describe the bridge rule (see (3) in Figure 1) that infers the degree of intention towards a goal φ for each plan α that allows to achieve the goal ($I_\alpha \varphi$), in the next section 3.

In order to represent and reason about graded mental attitudes we use a modal many-valued approach where reasoning about graded uncertainty, preferences and intentions is dealt with by defining suitable modal theories over suitable many-valued logics. The formalization of the adequate logics (i.e. language, semantics, axiomatization and rules) for the different contexts and the basic bridge rules can be seen in [1].

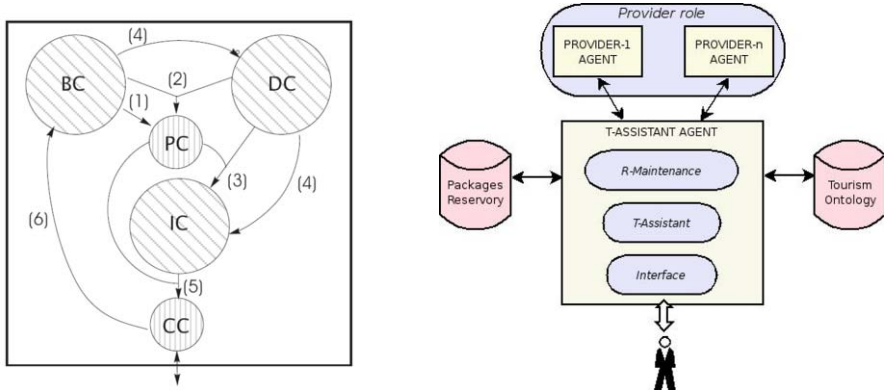


Figure 1. Multi-context model of a graded BDI agent (left) and the multiagent architecture of the Tourism Recommender system (right)

3. A Tourism Recommender System

For a prototype version of the Tourism Recommender System, we define two agent's types: provider agents and a Travel Assistant agent. As it is natural in the tourism chain, different tourist operators may collaborate/compete for the provider role. To represent these different sources of tourist packages, we use for this implementation two different agents (*P-Agents*), but the multiagent architecture is easily scalable to include other providers. These agents are only considered as packages suppliers and therefore, we do not get into their internal architecture. The agents in the Recommender system with the main source of information they interact with (i.e., the destination ontology and the package repository) are illustrated in Figure 1 (right). The implementation of the Recommender system was developed using a multi-threaded version of Prolog² allowing an independent execution of different contexts (i.e. in different threads). The principal role of the Travel assistant agent (*T-Agent*) is to provide tourists with recommendations about Argentinian packages and it can be suitably modelled as an intentional agent and particularly, by a g-BDI model. This agent model is specified by a multicontext architecture having mental and functional contexts. Next, we briefly describe how the contexts have been implemented in order to obtain the desired behaviour of the *T-agent* (for a detailed description see [3]).

Communication Context (CC): The CC is the agent's interface and is in charge of interacting with the tourism operators (*P-Agents*) and with the tourist that is looking for recommendation. The *T-Agent*, before beginning its recommendation task, updates its information about current packages (carrying out its repository maintenance role). It behaves as a wrapper translating the incoming packages into the *T-Agent* format and sends them to the Planner context. The user interface has been developed as a Web service application³ and it is responsible for:

- *Acquiring user preferences:* they are explicitly obtained from the user by filling in a form. The tourist can set her preferences (positive desires) and restrictions (negative desires) and assign them a natural number from 1 to 10 to represent the level of preference (resp. restriction) for the selected item. Preferences are given about the following issues: geographic zone, natural resources, infrastructure, accommodation, transport

²<http://www.swi-prolog.org>

³In this prototype the interface is in Spanish, an English version is ongoing work.

or activities. The constraints are related to the maximum cost he is able to afford, the days available for traveling and the maximum total distance he is willing to travel. Once the user finishes his selection, the CC sends all the acquired information to the Desire context DC.

- *Showing the resulting recommendation:* as a result of the *T-Agent* deliberation process, the CC receives from the Intention context a ranking of feasible packages that satisfies some or all of the tourist preferences. Then, he can visualize the information about them (i.e. the description of the transport, destination, accommodation, activities) opening suitable files.

- *Receiving Tourist's feedback:* after analyzing the ranking of the recommended packages, the user can express through the CC interface her opinion about the recommendation. Namely, the user can select one of the following three possible evaluations:

1. *Correct:* when the user is satisfied with the ranking obtained.
2. *Different order:* when the recommended packages are fine for the user, but they are ranked in a different order than the user's own order. In such a case, the user is able to introduce the three best packages in the right order.
3. *Incorrect:* The user is not satisfied with the given recommendation. Then, the interface enables him to introduce a (textual) comment with his opinion.

All the information resulting from the user data entry is stored to evaluate the system behaviour.

Desire Context (DC): As the *T-Agent* is a *personal agent*, its overall desire is to maximize the satisfaction of the tourist's preferences. Thus, in this context the different tourist's graded preferences and restrictions are respectively represented as positive and negative desires. For instance, the preferences of a tourist that would like to go to a mountain place and to travel by plane but not more than 2000 kms could be represented by the following theory:

$$\mathcal{T}_{DC} = \{(D^+(resources, mountain), 0.9), (D^+(transport, air), 0.7), \\ (D^+[(resources, mountain), (transport, air)], 0.92), (D^-(distance, 2000), 0.5)\}$$

The *T-Agent* uses the desires as pro-active elements, and are passed by a bridge rule to the Planner context that looks for feasible packages.

Belief Context (BC): In this context the *T-Agent* represents all the necessary knowledge about tourism and the Argentinian domain: tourist packages (each package is represented as a list containing an identifier, a tour provider, the package cost and a travel-stay sequence), information about destinations (represented by a destination ontology) and rules to infer how much preferences can be satisfied (to some degree) by the feasible tourist packages. This context also contains knowledge about *similarity relations*⁴ between concepts to extend the possibility of satisfying a tourist with similar preferences than the actually selected ones. Besides, the BC is in charge of estimating the extent (the belief) $B([\alpha_P]\varphi)$ to which a desire (preference) φ will be achieved when selecting a given package α_P .

Planner Context (PC): This context it is responsible for looking for *feasible packages*. A package is *feasible* when it satisfies at least one of the positive desires and its

⁴They are application domain dependent and more details can be seen in [3].

execution does not violate any restriction. These plans are computed within this context using an appropriate search method, that takes into account beliefs and desires injected by bridge rules from the BC and DC units, respectively. This set of packages is passed to the Intention context which is in charge of ranking them.

Intention Context (IC) and a Bridge rule example: In the IC the *T-Agent* finds by using a bridge rule the intention degree $I_\alpha\varphi$ for each feasible package α that is expected to satisfy a desire φ . This value is computed by a function f that suitably combines factors like the degree d of desire about φ , the belief degree r in achieving φ by executing the plan α , and the (normalized) cost c of the plan α .

$$\frac{DC : (D^+\varphi, d), BC : (B[\alpha]\varphi, r), PC : fplan(\varphi, \alpha, P, A, c)}{IC : (I_\alpha\varphi, f(d, r, c))} \quad (BR1)$$

Different functions can model different individual agent behaviours. In the *T-Agent* this function is defined as a weighted average: $f(d, r, c) = (w_1 * d + w_2 * r + w_3 * (1 - c)) / (w_1 + w_2 + w_3)$, where the different weights w_i are set by the *T-Agent* according to the priority criterion selected by the user (minimum cost or preference satisfaction). Once the rule has been applied to all the feasible plans, the IC has a set of graded intention formulae. Using the intention degrees the *T-Agent* makes a package ranking that communicates to the CC and then, through the user interface, it is provided back to the user as recommendation.

4. Validation of the Recommender Agent

It has been shown in [2,3] that the g-BDI architecture described above is useful to model recommender systems. In this section we try to answer whether this system provides satisfactory recommendations. This recommender system is accessible via Internet⁵ allowing an online and a multiuser access. To analyze its behaviour the user opinion is crucial. We want to know whether the *T-Agent* is a personal agent satisfying, to some degree, a set of different users.

We use the implementation of the *T-Agent* modelled as a g-BDI agent and a set of 40 tourism packages offered to the *T-Agent* by the provider agents. We have collected a set of 52 queries made to at least 30 different users. The preferences and restrictions introduced by them as input to the system, together with the system results and the user feedbacks, constitute our *N-cases* set. Each case in the dataset is composed by:

- **User Input:** a user ID and his graded preferences and restrictions.
- **Agent Result:** the system returns a ranking of at most nine packages.
- **User Feedback:** after analyzing the information of the recommended plans the user provides a feedback by evaluating the result as: (1) Correct, (2) Different order or (3) Incorrect.

Results: In this validation process, among the the selected 52 cases⁶ (*N-cases*), we have considered important to classify cases either as *satisfactory* (*S-Cases*) when the feedback type is (1) or (2), since the user can find what he wants among the recommended options,

⁵<http://musje.iiia.csic.es/eric/>

⁶Actually, for the validation purposes, we have only taken into account those cases which included the user's feedback.

or as incorrect ones otherwise. The cases where the user provides his own ranking (those with feedback type (2): Different order) are indeed very valuable because it means that the user analyzed the offers proposed by the system, while cases with the feedback type (1) (Correct order) sometimes they correspond to a “quick answer”. The results obtained for the N -cases classified by the different feedback categories are shown in the following Table. From these results, the global behaviour of the T -Agent may be considered useful in most cases (73% of N -cases).

| Queries (N -cases) | Correct order | Different order | Satisfactory (S -cases) | Incorrect |
|-----------------------|---------------|-----------------|----------------------------|-----------|
| 100% | 40.4% | 32.7% | 73.1% | 26.9% |

In order to give a general measure of the T -Agent results over the satisfactory cases (S -cases), we have evaluated how close is the T -Agent ranking with regard to the user own ranking. For this, we choose the Block (Manhattan) distance between the position of the first three packages selected by the user and their position in the system ranking. This distance was adopted because it is appropriate for capturing positional differences. Namely, assume the user feedback is $U_i = (P_{i1}, P_{i2}, P_{i3})$ and the T -Agent ranking for this consult is $R_i = (R_1, R_2, \dots, R_9)$. Then, if $P_{i1} = R_j$, $P_{i2} = R_k$, $P_{i3} = R_n$, the distance between the user and the system rankings is defined by:

$$Dist(U_i, R_i) = |1 - j| + |2 - k| + |3 - n|$$

The frequencies of the Block distance corresponding to the T -Agent results for all the S -cases can be seen in Figure 2. We analyzed the incorrect cases and the comments attached (if any) about the user dissatisfaction with respect to the system recommendation and they were somewhat scattered. Apart from that, in some of these incorrect cases we detected a system shortcoming related to the tourism knowledge base, the destination ontology used for this experimentation was incomplete with respect to the popular knowledge. Therefore, we believe the T -Agent behaviour may be improved by completing these ontologies. Finally, the S -cases set of satisfactory results (see table in Figure 2) yields an average distance of 2.95 in the scale $[0, 18]$, and hence giving a good global measure result. Summarizing, we have obtained satisfactory results of the Recommender System in this validation process that allows us to claim that “the T -Agent recommended rankings over Tourism packages are in most cases near to the user own rankings”.

5. Experimentation

In this section we present the experimentation we have made following two directions. The first one, we call it Sensitivity Experimentation, has the purpose of analyzing how much the general g-BDI agent architecture can model concrete agents having different behaviours by modifying some of its components. The second one aims at checking whether the distinctive feature of the g-BDI agent model, which is the gradual nature of mental attitudes, actually makes a difference (in terms of better results) with simulated BDI non-graded models.

5.1. Sensitivity model experimentation

We have performed two experiments to analyze how the overall recommender system behaviour can be modified by tuning some of the T -agent components. First, in *Experiment 1* we change the theory of the desire context DC, using another way for computing

the desire degree for each preference combination. Then, in *Experiment 2* we modify the bridge rule (*BR1*) definition by changing the function f to obtain the intention degree.

Experiment 1 For this experiment we follow the next steps:

(1) We use the tourism recommender agent *T2-Agent*: this agent was developed changing the DC in the *T-Agent*. The modification in this context is related to the way the desire degrees are computed. The underlying idea was to weight not only the preference degree but also the number of preferences we are considering in each combined desire, as to give more relevance to the desires that combine a higher number of preferences. For this purpose in the Desire Context of the *T2-Agent* we use as degree for desire D the value

$$d' = 1/2 * (d + \frac{CardD}{CardPref})$$

where d is the degree used in the *T-Agent*, $CardD$ is the number of preferences considered in the desire D and $CardPref$ is the number of preferences selected by the user.

(2) We consider the *S-cases* (see Validation 1) where the results were satisfactory.

(3) The *user inputs* of the *S-cases* are run in the *T2-Agent*.

(4) We compare the *T2-Agent* results with the *S-cases user feedbacks* we have for the *T-Agent* and compute distances.

Results: In the experiment we compare the ranking proposed by *T2-Agent* with the feedback of the *S-cases* consisting of the first three packages extracted from the *T-Agent* recommendation. Some of these packages may not be found in the *T2-Agent* answer. As in the validation process, we use the Block distance to have a global measure of the *T2-Agent* performance. For the missing packages, we take an optimistic approach assuming that the distance is 10 (supposing that the missing packages would be in the first place immediately after those appearing in the ranking). The distance frequency corresponding to the *T2-Agent* results for all the *S-cases* are shown on the table and the graph of Figure 2. For this experiment we use two global measures, the *average* of distances excluding those cases having missing packages and the *total average* of distances including all the cases. The first one is a measure of how much the agent captures the user's own ranking, while the second one measures the dispersion of the agent results. The average of the distances between the *T2-Agent* ranking and the feedback of the *S-cases* is 2.85 and the total average is 4.23. Comparing the first global measure with the one obtained with the *T-Agent* results, we notice that this measure is slightly better than the one obtain for the *T-Agent* results. We think that in a direct measure of the performance of the *T2-Agent* (comparing the *T2-Agent* ranking with the corresponding user feedback) we would have had better results. Thus, we can conclude that *T-Agent* and *T2-Agent* share a similar behaviour, but the *T2-Agent* results are a little bit closer to the user selections.⁷

Experiment 2 We follow the same steps used in *Experiment 1* but in this case, for item

(1) we use a tourism recommender agent called *T3-Agent*. This new agent has been defined from the *T-Agent* by changing one of its bridge rules. Namely, we have modified bridge rule (*BR1*) (see Section 3) that computes the intention degree of a package α in order to satisfy a set of user preferences φ . We have used for *T3-Agent* a function that

⁷ Alternative measures could have been implemented, like the percentage of cases that are below a certain distance as kindly suggested by one referee.

assigns an intention degree according to two different priorities (Preference Satisfaction or Minimum Cost) by defining two lexicographic orderings, as follows:

- when the *Preference Satisfaction* criterion is selected, we consider the intention $I_{\alpha}\varphi$ described by the 3-tuple $(d', r, 1 - c)$, where d' is the desire degree of φ , r is the belief degree in satisfying the user preferences φ by the considered plan α , and c is the cost of the plan α . Then, we use the lexicographic order on the product space $[0, 1]^3$ to rank the 3-tuples and hence the intentions.
- when the *Minimum Cost* criterion is selected, we consider $I_{\alpha}\varphi$ described by the 3-tuple $(d', 1 - c, r)$ and then we rank intentions by lexicographically ordering these tuples.

Results: The resulting distance frequencies of the results of the *T3-Agent*, for the *S-cases* are shown in Figure 2 (compared with the ones obtained by *T-Agent* and *T2-Agent*). The average of the distances in this case was 4.97, worse than the previous experiments, and the total average (including cases with missing packages) is 6.73. This means that the ranking obtained by this *T3-Agent* is farther from the user ranking obtained in the validation process than the results of the previous versions *T-Agent* and *T2-Agent*. This result may be interpreted as indicating that indeed the way of computing the intention degree does have an influence in the behavior of the recommender agent. Thus, we can state that the g-BDI model allows us to engineer recommender agents having different behaviors by tuning the different components of the architecture.

| Distance | Frequency | | |
|-------------|-----------|----------|----------|
| | T-Agent | T2-Agent | T3-Agent |
| 0 | 21 | 11 | 5 |
| 1 | 1 | 4 | 3 |
| 2 | 2 | 4 | 3 |
| 3 | 3 | 5 | 3 |
| 4 | 2 | 1 | 2 |
| 5 | 0 | 1 | 2 |
| 6 | 2 | 4 | 3 |
| 7 | 2 | 2 | 0 |
| 8 | 0 | 0 | 3 |
| 9 | 0 | 0 | 2 |
| 10 | 0 | 1 | 3 |
| 11 | 3 | 2 | 1 |
| 12 | 0 | 1 | 4 |
| 13 | 0 | 1 | 1 |
| 14 | 1 | 0 | 1 |
| 17 | 1 | 0 | 0 |
| 18 | 0 | 0 | 1 |
| 26 | 0 | 0 | 1 |
| 30 | 0 | 1 | 0 |
| Average | 2.95 | 2.85 | 4.97 |
| Tot.Average | 2.95 | 4.23 | 6.73 |

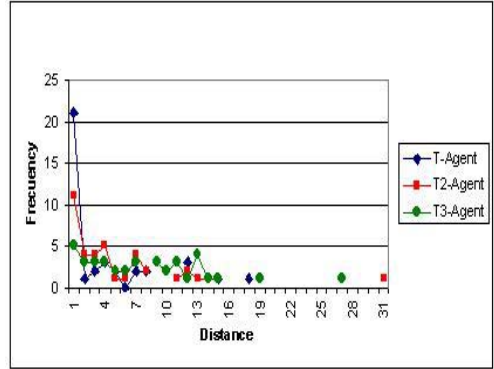


Figure 2. Distance frequencies: table (left) and graph (right)

5.2. Graded vs. non-graded model comparison

The aim of this experimentation is to compare the g-BDI model with non-graded (two-valued) BDI architectures. We want to show that the graded model of agent allows us to implement recommender agents having better results than the ones based on non-graded BDI models. We use the *T-Agent* and *T2-Agent* prototypes as g-BDI model implementations. Since the development of a Tourism Recommender using another traditional BDI architecture would be a time-demanding task and since also different factors would possibly interfere in the comparison of the results (e.g. how the agent builds plans, which decision process she uses), for simplifying and clarifying purposes we have decided to use simulated non-graded versions of the BDI architecture of the tourism agent. Start-

ing from the recommender agents *T-Agent* and *T2-Agent* we keep their multicontext architecture and their logic schemes for contexts⁸. Then we introduce some thresholds to make the desire and belief attitudes two-valued (i.e. their degrees will be allowed to only take values in $\{0, 1\}$). The intention degrees have been left many-valued as to obtain a ranking of the selected packages.

Experiment 3 We have followed the same procedure as for the previous ones but, for this case, we use a family of Tourism Recommender agents called *Cij-Agents*. These agents derive from the recommender *T-Agent* or *T2-Agent* and simulate two-valued models of BDI agents. Each *Cij-Agent* has been developed by introducing thresholds in the context DC (U_d) and in the context BC (U_b) of the *T-Agent* and *T2-Agent*, to decide which formulae in these contexts are considered to hold (i.e. those with degree 1) and which do not (i.e. those with degree 0). Then, the following internal processes are introduced in these contexts:

- **DC:** before introducing formulae like $(D^+\phi, d)$ in the DC it is checked whether $d \geq U_d$; if so, the formula $(D^+\phi, 1)$ is added in the context, otherwise this desire is discarded (supposing $(D^+\phi, 0)$).
- **BC:** the same happens when the belief context evaluates the degree r of formulae like $(B[\alpha]\varphi, r)$, if $r \geq U_b$ then the formula $(B[\alpha]\varphi, 1)$ is added to the BC, otherwise its degree is considered to be 0.

As for the setting of the different thresholds, we have analyzed the desire and belief degrees distribution in the *T-Agent* and *T2-Agent* previous executions. We have experimented different thresholds and it turns out that with three different values (0.4, 0.5 and 0.6) we can obtain a good representation of the whole variations in the agents' results. Then, we have defined the following "two-valued" BDI agents using three thresholds:

- Deriving from *T-Agent*:

1. *C14-Agent* uses $U_d = U_b = 0.4$
2. *C15-Agent* uses $U_d = U_b = 0.5$
3. *C16-Agent* uses $U_d = U_b = 0.6$

- Deriving from *T2-Agent*:

4. *C24-Agent* uses $U_d = U_b = 0.4$
5. *C25-Agent* uses $U_d = U_b = 0.5$
6. *C26-Agent* uses $U_d = U_b = 0.6$

Then, we have run the *S-cases* in each agent of these two-valued families and compared the results with the *S-cases* feedback computing the Block distances. As in the previous experimentation we have used the distance average and the total average as global measures.

Results: The resulting average distance and total average (including the cases with missing packages) of the results of these two families of crisp agents deriving from *T-Agent* and *T2-Agent* are gathered in the following table:

| | <i>T-Agent</i> family | | <i>T2-Agent</i> family | |
|-------------------|-----------------------|--------------|------------------------|--------------|
| | Average | Tot. Average | Average | Tot. Average |
| g-BDI model | 2.95 | 2.95 | 2.85 | 4.23 |
| $U_d = U_b = 0.4$ | 6.43 | 14.06 | 4.04 | 8.36 |
| $U_d = U_b = 0.5$ | 6.43 | 14.83 | 3.50 | 8.07 |
| $U_d = U_b = 0.6$ | 4 | 17.41 | 3.55 | 14.43 |

⁸This is possible as the many-valued frameworks used for the mental contexts are extensions of classical logic used in the two-valued models.

Comparing the averages obtained with the two-valued models of recommenders (deriving from *T-Agent* and *T2-Agent*) we can see that those corresponding to the thresholds 0.4 and 0.5 are very similar. The average achieved with the threshold 0.6 is the best in the *T-Agent* family and is almost the best in the *T2-Agent* one, but the total average is greater, meaning that we have more packages of the *S-cases* feedback out of the system ranking. In both families we can see that the distance average and the total average of the recommenders using graded models are better than the simulated two-valued ones (using three different thresholds). These results give support to the claim that the recommender agents modelled using graded BDI architectures provide better results than the ones obtained using two-valued BDI models.

6. Conclusions

In this work we have focused on the validation and experimentation of g-BDI agents using as a case study a Tourism recommender agent. First, the results of the validation performed allow us to conclude that g-BDI agents are useful to build recommender systems in a real domains such as tourism, providing satisfactory results. Second, we have also performed a sensitivity analysis showing that a g-BDI agent architecture can engineer concrete agents having different behaviours by suitably tuning some of its components. How to select the most suitable recommender agent (from the alternative presented) according to a more detailed user profile is out the scope of this paper and is left as future work. Finally, the results of a third experiment support our claim that the distinctive feature of recommender systems modelled using g-BDI agents, which is using graded mental attitudes, allows them to provide better results than those obtained by non-graded BDI models.

Acknowledgments: The authors thank the anonymous reviewers for their helpful remarks. L. Godo and C. Sierra acknowledge partial support of the Spanish project Agreement Technologies (CONSOLIDER CSD2007-0022, INGENIO 2010).

References

- [1] Casali A., Godo L. and Sierra C. Graded BDI Models For Agent Architectures. In Leite J. and Torroni P. (Eds.) *CLIMA V*, LNAI 3487, pp. 126-143, Springer 2005.
- [2] Casali A., Godo L. and Sierra C. Modeling Travel Assistant Agents: a graded BDI Approach. In Proc. of IFIP-AI, WCC. Artificial Intelligence in Theory and Practice., Max Bramer (Ed.) (ISBN 0-387-34654-6), pp. 415-424 Springer Verlag, 2006.
- [3] Casali A., Von Furth A., Godo L. and Sierra C. A Tourism Recommender Agent: From theory to practice. Proc. of WASI-CACIC 2007 (ISBN 978-950-656-109-3) pp. 1548-1561, Corrientes (Argentina) 2007.
- [4] Ghidini C. and Giunchiglia F. Local Model Semantics, or Contextual Reasoning = Locality + Compatibility. *Artificial Intelligence* 127(2), pp. 221-259, 2001.
- [5] Montaner M., Lopez B. and de la Rosa J. L. A Taxonomy of Recommender Agents on the Internet. *Artificial Intelligence Review* 19 (4), pp. 285-330, 2003.
- [6] Niinivaara, O., Agent-Based Recommender Systems. Technical Report, University of Helsinki, Dept. of CS, 2004.
- [7] Rao, A. and Georgeff M. BDI agents: From theory to practice. In *proceedings of the 1st International Conference on Multi-Agents Systems*, pp. 312-319, 1995.
- [8] Ricci F. Travel recommender Systems. *IEEE Intelligent Systems* 17 (6), pp. 55-57, 2002.

Agent Negotiation Dissolution

Nicolás HORMAZÁBAL^a, Josep Lluís DE LA ROSA I ESTEVE^a and
Silvana ACIAR^a

^a*Agents Research Lab
Universitat de Girona (UdG)
E17071 Girona (Catalonia)*

Abstract. In Multi-Agent Systems, agents negotiate between them for coordination and collaboration, but their preferences and the autonomous behaviour of each participant agent can lead the negotiation through several steps before they find a common agreement. As every agent in a negotiation isn't aware about the other's preferences or about their decision making process, the negotiation workflow needs to go through an intense communication process that not always ends with the most desirable agreement between them.

In this paper we propose a recommender system that suggests the best moment to end a negotiation. The recommendation is made from a trust evaluation of every agent in the negotiation based on their past negotiation experiences. For this, we introduce the Trust Aware Negotiation Dissolution algorithm.

Keywords. Multi-Agent Systems, Negotiation, Dissolution, Trust, Coalitions

1. Introduction

Negotiation and cooperation are critical issues in multi-agent environments [4], such as in Multi Agents Systems and research on Distributed Artificial Intelligence. In distributed systems, high costs and time delays are associated with operators that make high demands on the communication bandwidth [1].

Considering that agents are aware of their own preferences, which help their decision making during the negotiation process, the negotiation can go through several steps depending on their values as each agent does not know the others' preferences. This can lead to an increase of communication bandwidth costs affecting the general performance, and might put agents in undesirable negotiation situations (such as a negotiation that probably will not end with an acceptable agreement).

Termination of the negotiation process or a negotiation dissolution action should be considered when the negotiation is in a situation where the expected result of the following steps cannot be better than the current result. This will not only help to determine when to end a negotiation process, but also to help decide whether to end it with or without an agreement.

This paper suggests a recommender algorithm for finding the best moment to end a negotiation between agents. In section 2 we introduce the negotiation dissolution, in section 3 we explain the Trust Aware Negotiation Dissolution algorithm applied for the negotiation, in section 4 we will present an example to explain how the algorithm works

and in section 5 we will show our results on experimentation. Then in section 6 we will present our conclusions and the related future work.

2. Negotiation Dissolution

Not every negotiation process should end with an agreement, or at least with an acceptable one for every participant. During a negotiation, a problem arises when an agent has to decide if he accepts the offer, refuses it making a counter-offer or simply ends the negotiation process without an agreement. How is possible to know if the received offer is the best one an agent can expect from the bidder. Let's consider the following example, through a simple bargaining negotiation process, where the communication workflow goes on terms of offers and counter-offers between the participants according to the negotiation constraints.

Two peasants negotiate for a barter. The first one offers two bags from his harvest; one full of corn and another full of rice in exchange of two bags of wheat. The other one thinks that is an unfair exchange as he already has a lot of rice and giving so much wheat for a bag of corn doesn't seem the best trade for him, so makes a counter-offer of one and a half bag of wheat for the corn and the rice. They can continue negotiating until they find a common acceptable agreement. But the case is that sometimes, the best offer is not always the last one, maybe at some step the first peasant arrives to think that he should have accepted the offer of one and a half bag of wheat from before, and now it seems impossible to find a better offer. Also it is possible that they will not find an acceptable agreement at all, after several negotiation steps and it would have been better to end the negotiation before.

This is what we call the Negotiation Dissolution; an early ending of the negotiation that is not driven by the agent's decision making process.

3. Trust Aware Negotiation Dissolution

The Trust Aware Negotiation Dissolution algorithm (TAND from now on) takes into account direct interactions from similar situations in the past (Situational Trust [5]). The situation is defined by the current negotiation status. A simple situation α , can be defined by the n issues i from the negotiation offer:

$$\alpha = \{i_1, i_2, \dots, i_n\} \quad (1)$$

An autonomous agent that is capable to make flexible and complex negotiations, should consider three broad areas [6]: What is the negotiation protocol, what are the issues over which the negotiation takes place and what is the reasoning model the agents employ. These areas will affect the way an agent classifies the situations and calculates the similarities for finding past situations.

This situation value α will be used to calculate a situational trust, which is the amount of trust that one agent (partner in the negotiation) has in another taking into account a specific situation [5]. The basic formula (based on the original approach on [5]) used to calculate this type of trust is:

$$T_a(y, \alpha) = U_a(\alpha) \times T_a(\widehat{y}, P_\alpha) \quad (2)$$

Where:

- a is the evaluator agent.
- y is the target agent.
- α is the situation.
- $U_a(\alpha)$ represents the utility that a gains from a situation α , calculated by its utility function.
- P_α is a set of past situations similar to α .
- $T_a(\widehat{y}, P_\alpha)$ is an estimated general trust for the current situation. We will calculate this value considering two possible results for each situation in the set of past interactions P_α , that are similar to α : a successful result or an unsuccessful one (whether or not an agreement was reached). This leads to the calculation of the probability that the current situation could end in an agreement based on past interactions (based on the work in [9]). This is calculated by:

$$T_a(\widehat{y}, P_\alpha) = \frac{e}{n} \quad (3)$$

Where e is the number of times that an agreement has been made with the agent y on the each situation from P_α , and n is the total number of observed cases in P_α with the agent y . $n = |P_\alpha|$.

A function g based on agent a 's decision process returns the set S of j possible negotiation situations (the offers the agent is willing to make) σ based on the current situation α the agent is in:

$$g : \alpha \longrightarrow S \quad (4)$$

$$S = \{\sigma_1, \sigma_2, \dots, \sigma_j\} \quad (5)$$

From the possible situations, we obtain the *best expected situational trust* $E_a(y, S)$; which obtains the trust for the best expected case from among the possible situations in which the agents can find themselves in the future, given the current negotiation:

$$E_a(y, S) = \max_{\sigma \in S} T_a(y, \sigma) \quad (6)$$

Having the trust in the current situation $T_a(y, \alpha)$, and the best expected situational trust $E_a(y, S)$, we can calculate a rate that will help the agent decide whether or not they should continue the negotiation. The situational trust at the present time, divided by the potential situational trust gives us the Dissolution Rate R , which in conjunction with a minimum acceptable trust value M , will help to decide whether or not to dissolve the negotiation process.

$$R = \frac{T_a(y, \alpha)}{E_a(y, S)} \quad (7)$$

The dissolution decision depends on the value of R :

Table 1. Negotiation example with possible agreements and their agents preferences.

| D | Agents | |
|-----|-----------|-----------|
| | f_{A_1} | f_{A_2} |
| 1 | 0.6 | 0.3 |
| 2 | 0.8 | 0.4 |
| 3 | 0.2 | 0.9 |
| 4 | 0.3 | 0.5 |

$$\begin{aligned}
R &\geq 1 \Rightarrow \text{Dissolve} \\
(R < 1) \vee (E_a(y, S) < M) &\Rightarrow \text{Dissolve} \\
(R < 1) \vee (E_a(y, S) \geq M) &\Rightarrow \text{Continue Negotiating}
\end{aligned} \tag{8}$$

In other words, if, based on future steps, the expected situation does not have a better trust value than the current one, the best thing to do is to end the negotiation now. Otherwise, it is better to continue negotiating.

4. Example of a Negotiation Dissolution

We will present a negotiation case as an example, where two agents negotiate for reaching an agreement from a limited number of options, each agent with its personal preference for each possible agreement.

4.1. Case 1: Plain Negotiation Dissolution

Let's consider two agents A_1 and A_2 , in a negotiation with a set D of n possible agreements d such as:

$$D = \{d_1, d_2, \dots, d_n\} \tag{9}$$

Each agent has an utility function f that returns the agent a 's preference p for each agreement $d \in D$. Example values are in Table 1.

$$f_a : D \longrightarrow P \tag{10}$$

$$P = \{p_1, p_2, \dots, p_n\} \tag{11}$$

For this example, the decision making process consists in offering the best evaluated agreement (the one which utility function has the higher value), and counter-offer the next best one until it has no better agreement than the offered one. In this case, agent A_1 will first offer the option 2, agent A_2 will counter-offer the option 3, and so on until they find an agreement. If a message is sent as $Send(sender, receiver, agreement)$, the message sequence between the agents should be as shown on table 2.

At the fourth step, the agent A_1 realizes that his next best option is the agreement 3, but is not better than the offered one, so accepts the proposed agreement 4.

Table 2. Messages during the negotiation example.Step1 : $Send(A_1, A_2, 2)$ Step2 : $Send(A_2, A_1, 3)$ Step3 : $Send(A_1, A_2, 1)$ Step4 : $Send(A_2, A_1, 4)$

4.2. Case 2: Negotiation Dissolution with TAND

Repeating the same negotiation using TAND, if we consider as a unique situation each possible agreement in D as d_i , from equation 2.

$$d_i := \alpha \quad (12)$$

In our example we are considering only one past case. So the set P_α from equation 3 will contain only one value which belongs to the past recorded situation similar to α with its results. From the table 2, the result for the situations 1 and 2 is a rejected offer for an agreement (we understand a counter-offer as a rejection of the previous offer), and for the situations 3 and 4 are offered agreements from the other partner.

$$P_1 = \{0\}, P_2 = \{0\}, P_3 = \{1\}, P_4 = \{1\} \quad (13)$$

Instead of a unique trust general value, we are going to put these values into a vector, where each row represents a situation. Using all the past cases P_i for each situation contained in the set P , if we are taking only one past case ($n = 1$), the agent A_1 's estimated general trust vector for each situation for agent A_2 is:

$$T_{A_1}(\widehat{A_2}, P) = \begin{pmatrix} 0/1 \\ 0/1 \\ 1/1 \\ 1/1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \end{pmatrix} \quad (14)$$

This means that based on past interactions, agent A_1 has seen a 100% of good (or positive) cases in situations 3 and 4 with agent A_2 . As we saw in section 4.1, agent A_2 made offers for these possible agreements or situations.

Let's repeat the negotiation process described before, and stop at the step 2, when agent A_1 receives a counter-offer for the agreement 3. The trust for the current situation (an offer for the agreement 3) is:

$$\alpha = 3 \quad (15)$$

$$T_{A_1}(A_2, \alpha) = U_{A_1}(\alpha) \times T_{A_1}(\widehat{A_2}, P_3) \quad (16)$$

$$T_{A_1}(A_2, \alpha) = 0.2 \times 1 = 0.2 \quad (17)$$

As the decision making process suggests the agent to consecutively make offers from the one with higher preference value to the lower one, the expected future situations are (it already made an offer for the agreement 2):

$$S = \{1, 4, 3\} \quad (18)$$

Which utilites U will be (by its utility function f), and their estimated general trust $T_a(\widehat{y}, \alpha)$:

$$U_{A_1}(\alpha)_{\alpha \in S} = (0.6 \ 0.3 \ 0.2) \quad (19)$$

$$T_{A_1}(\widehat{A_2}, \alpha)_{\alpha \in S} = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix} \quad (20)$$

$$U_{A_1}(\alpha)_{\alpha \in S} \times T_{A_1}(\widehat{A_2}, \alpha)_{\alpha \in S} = (0 \ 0.3 \ 0.2) \quad (21)$$

Then:

$$E_{A_1}(A_2, S) = \max(0 \ 0.3 \ 0.2) = 0.3 \quad (22)$$

From equation 7, we get R :

$$R = \frac{T_{A_1}(A_2, \alpha)}{E_{A_1}(A_2, S)} = \frac{0.2}{0.3} \simeq 0.67 \quad (23)$$

From equation 8, we could suggest to continue on the negotiation as the expected result could be better than the current one as the R value is below 1, but also the best expected situational trust ($E_{A_1}(A_2, S)$) should be compared to the minimum acceptable trust value M also from the equation 8. In this case, is on the agent's criteria to evaluate if the expected trust value fulfills its expectations on the current negotiation.

5. Experiment and Results

For testing the TAND algorithm, we implemented a negotiation environment where two agents negotiate to reach an agreement from a limited number of options; agents consecutively offer their next best option at each step until the offer is no better than the received one. The scenario consists of different agents that each represent a person who wants to go to a movie with a partner, so they negotiate between them from different available movie genres to choose which movie to go to together. The environment was developed in RePast ¹.

5.1. Assumptions

In order to avoid affecting the system performance, agents will only save the trust of a limited number of potential partners in their knowledge base; that is, they will maintain a limited contact list instead of recording the experience of every partner they have negotiated with.

There will be a fixed number of available movie genres (for example, drama, comedy, horror, etc.) during the whole simulation.

¹<http://repast.sourceforge.net>

The similar past situations will be the past negotiations on the same movie genre than the current one.

Each agent will have a randomly generated personal preference value (from a uniform distribution) for each genre between 0 and 1, where 0 is a genre it does not like at all, and 1 is its preferred movie genre. One of these genres, randomly chosen for each agent, will have a preference value of 1, so each agent will have always a favorite genre.

We assume that there is always a movie in the theaters available for each genre. Each movie genre will be used to identify the situation α the negotiation is in, for the calculation of the trust from equation 2.

The result of the utility function $U_a(\alpha)$ will be the preference for each movie genre randomly assigned for each agent.

Partners involved in the negotiation will be randomly chosen.

An agent can participate only in one negotiation at one time.

The experiment will run through three different cases, each one with 100 agents and 10 different movie genres:

- Case 1: Contact list of 20 most trusted partners.
- Case 2: Unlimited contact list size.
- Case 3: No TAND, simple negotiation.

Every experiment will run through 2000 steps. At each step, 1/4 of the total population (25 agents for the cases described above) will invite another partner to a negotiation for a movie.

For evaluating the performance, we will use three values:

- Average steps used for all agreements made: AS (lower is better).
- Average preference (the final preference value during the agreement for each successful negotiation): AP (higher is better).
- Average distance from the perfect pitch: AD (lower is better).

We define the perfect pitch P as the highest value for the product of each agent a in the A set of participating agents' preference (result of the utility function $U_a(\alpha)$ for each movie genre) from every possible agreement d :

$$P = \max_{d \in D} \left(\prod_{a \in A} f_a \right) \quad (24)$$

The distance from the perfect pitch is the difference from the negotiation preference K with the perfect pitch P .

$$AD = P - K \quad (25)$$

5.2. Results

After 20 experiments for each case, in every case at each experiment we averaged the results obtained, seen in table 3.

The results improve in cases 1 and 2, in terms of average steps AS needed for closing a negotiation with an agreement, compared to case 3, where TAND is not used. However, the average preference AP has a higher value, and the distance from the perfect pitch AD is reduced more than 35% from case 3 to case 2. The contact list size is a critical

Table 3. Average Final Steps.

| | | Case 1 | Case 2 | Case 3 |
|----|---------|--------|--------|--------|
| AS | Avg | 5.2894 | 4.6683 | 5.6993 |
| | Std Dev | 0.0283 | 0.0249 | 0.0282 |
| AP | Avg | 0.8001 | 0.8168 | 0.7892 |
| | Std Dev | 0.0073 | 0.0064 | 0.0080 |
| AD | Avg | 0.1370 | 0.1125 | 0.1548 |
| | Std Dev | 0.0034 | 0.0030 | 0.0048 |

issue, as we can see from comparing results between cases 1 and 2, that the improvement is higher when there are no limits in the contact list's size.

As for the amount of rejected negotiations (negotiations closed without agreement), they are below 1% of the total negotiations.

6. Concusions and Future Work

We have presented TAND and its preliminary results, where we can see that it improves the negotiation process in terms of agents' preferences and number of steps to achieve an agreement. Taking into account possible agents' performance issues, a limited contact list should be considered, but its size limitation could negatively affect the TAND results as we can see in table 3, so that work finding the optimal contact list size should be done. As far as now, the contact list filling criteria are simple, in the trusted contact list, agents with higher trust replace the agents with lower values and when the contact list is full, improved results are expected using other criteria for dealing with the contact list, for example using different levels of priorities, or a relation with the partner selection criteria (in the experiments the selection is made randomly).

TAND has been tested on a simple bilateral negotiation process, but can also be used on other types of temporary coalitions such as dynamic electronic institutions [7] for supporting their dissolution phase. Future work will focus on this, expanding its scope to generic types of coalitions. In addition, work on implementing other ways to calculate trust should be done, and other methods to manage the dissolution (such as Case Based Reasoning [2]) in order to compare results. The topic of dissolution of coalitions is not a new one, but it is not a topic that has been studied in depth [2], so this research topic provides a wide open field that needs to be explored.

References

- [1] H. Bui, D. Kieronska, and S. Venkatesh, 'Learning other agents preferences in multiagent negotiation', *Proceedings of the National Conference on Artificial Intelligence (AAAI-96)*, 114–119, (1996).
- [2] N. Hormazabal and J. L. De la Rosa, 'Dissolution of dynamic electronic institutions, a first approach: Relevant factors and causes', in *2008 Second IEEE International Conference on Digital Ecosystems and Technologies*, (February 2008).
- [3] N. R. Jennings, S. Parsons, C. Sierra, and P. Faratin, 'Automated negotiation', *Proceedings of the Fifth International Conference on the Practical Application of Intelligent Agents and Multi-Agent Technology*, 23–30, (2000).

- [4] S. Kraus, 'Negotiation and cooperation in multi-agent environments', *Artificial Intelligence*, 79–97, (1997).
- [5] S. Marsh, *Formalising Trust as a Computational Concept*, Ph.D. dissertation, Department of Mathematics and Computer Science, University of Stirling, 1994.
- [6] H.J. Mueller, *Negotiation principles*, 221–229, *Foundations of Distributed Artificial Intelligence, Sixth-Generation Computer Technology*, John Wiley and Sons, Inc., New York, 1996.
- [7] Muntaner-Perich and J.Ll. de la Rosa, 'Dynamic electronic institutions: from agent coalitions to agent institutions', *Workshop on Radical Agent Concepts (WRAC05), Springer LNCS (Volume 3825)*, (2006).
- [8] H.S. Nwana, Lee L., and N.R. Jennings, 'Co-ordination in software agent systems', **BT Technol J**, **14** N 4, 79–88, (1996).
- [9] M. Schillo, P. Funk, and M. Rovatsos, 'Using trust for detecting deceitful agents in artificial societies', *Applied Artificial Intelligence, (Special Issue on Trust, Deception and Fraud in Agent Societies)*, **14**, 825–848, (2000).

On Partial Deduction and Conversational Agents

Mariela MORVELI-ESPINOZA and Josep PUYOL-GRUART¹

Artificial Intelligence Research Institute (IIIA)

Spanish Scientific Research Council (CSIC)

Abstract.

Agents are situated autonomous entities that perceive and act in their environment, and communicate with other agents. An agent usually starts a conversation by querying another agent because it needs to satisfy a specific goal. This process allocates a new goal to the agent receiving the initial query, starting new dialogs with other agents, generating a recursive interaction. The generation of this kind of dialog is interesting when the system has the possibility of generating conditional answers with imprecise and uncertain values. We consider simple deliberative rule-based agents that proactively try to satisfy their goals. The mechanism to achieve this dialogs is based in the *specialization* of the mental state of agents, by means of the partial deduction of rule bases.

Keywords. Conversational agents, multi-agent systems, partial deduction, multiple-valued logic.

Introduction

Rule specialization has been used intensively in logic programming [14], mainly for efficiency purposes, but it has potential applications in other areas as multi-agent systems and particularly in communication among agents [11]. The proposal of this paper is not to explain the general advantages of an inference engine based on specialization [15, 16, 17], but to show that this mechanism is useful to drive the communication among agents, generating *reasonable* dialogs. We propose the use of this technique to model the communication behaviour between agents, in an uncertain context, by allowing agents to use *conditional answers* [7, 13].

In classical (boolean) rule bases, deduction is mainly based on the modus ponens inference rule: $a, a \rightarrow b \vdash b$. In the case that a denotes a conjunction of conditions $a_1 \wedge a_2$, the above inference rule is only applicable when every condition of the premise, i.e. a_1 and a_2 , is satisfied, otherwise nothing can be inferred. However, if we only know that condition a_1 is satisfied, we can use *partial deduction* to extract the maximum information from incomplete knowledge in the sense of the following *specialisation* inference rule: $a_1, a_1 \wedge a_2 \rightarrow b \vdash a_2 \rightarrow b$. The rule $a_2 \rightarrow b$ is called the *specialisation* of $a_1 \wedge a_2 \rightarrow b$ with respect to the proposition a_1 . The specialisation of a *rule base* consists

¹Corresponding author: Josep Puyol-Gruart, Artificial Intelligence Research Institute (IIIA). Campus UAB. 08193 Bellaterra. Spain. Tel.: +34 935809570; Fax: +34 935809661; E-mail: puyol@iiia.csic.es.

on the exhaustive specialisation of its rules. Rules will be substituted by its specialized versions, and rules with no conditions will be eliminated and new propositions will be added. These new propositions will be used again to specialize the agent. The process will finish when the agent has no rule containing on its conditions a known proposition.

In an approximate reasoning context the specialization is much more interesting. The above boolean specialization inference rule can be transformed in the following way: $(a_1, \alpha), (a_1 \wedge a_2 \rightarrow b, \rho) \vdash (a_2 \rightarrow b, \rho')$, meaning that if the proposition a_1 is known to be true at least to the degree α and the rule $a_1 \wedge a_2 \rightarrow b$ is true at least to the degree ρ , then the specialised rule $a_2 \rightarrow b$ is true at least to a degree $\rho' = f(\alpha, \rho)$, where f a suitable combination function.

Using conditional answers and the specialization mechanism, agents are able to answer, when needed, with the information the questioner should know to come up with a value for the query, or they may also inform about other deductive paths that would be useful to improve the solution [15]. For instance the agent can answer: *with the current information x is quite true, but if y were true then x will be definitively true.*

We will use a very simplified vision of agents as message passing entities containing rules. When an agent receives a query it starts a process of finding new external information in order to obtain an answer for that query. The difference with other approaches is that the agent will use the external information to specialize the knowledge base of the agent, and incrementally build more precise answers. The answer can be conditional, that is, it can contain rules if it is not possible to obtain enough information.

In Section 2 we formally describe both the agents and the specialization of their mental state. Section 3 is devoted to the description of the protocols. We present an example of conversation in Section 4. Finally, some discussion and the conclusions are developed in Section 5.

1. Mental state and specialization

The state of our agents will be their mental state [20]. The main component of the mental state is the knowledge base containing beliefs (facts) and knowledge (rules) for deliberation. In this Section a simplified version of our propositional language² and the inference mechanism will be described.

Definition 1 (Language and inference) $\mathcal{L} = \langle T_n, \Sigma, \mathcal{C}, \mathcal{S} \rangle$ is defined by:

- $T_n = \{t_0, t_1, \dots, t_n\}$ is an ordered set of truth-values, where t_0 and t_n are the booleans True (1) and False (0) respectively.
- Σ is a set of propositional variables (atoms or facts).
- \mathcal{S} are sentences composed by: atom pairs (a, V) , and rules of the form $(p_1 \wedge p_2 \wedge \dots \wedge p_n \rightarrow q, V)$, where $a, p_i, q \in \Sigma$, $V \in T_n$, and $\forall i, j (p_i \neq p_j, q \neq p_j)$

We will use the following inference rules:

- Parallel composition: from (φ, V_1) and (φ, V_2) infer $(\varphi, \max(V_1, V_2))$

²In the complete version of the language we consider negation and the values of facts and rules are intervals of truth values. For the sake of simplicity here we use *min* and *max* operations instead of general triangular norms. For more information please see [16].

- **Specialization:** from (p_i, V) and $(p_1 \wedge \dots \wedge p_n \rightarrow q, W)$ infer $(p_1 \wedge \dots \wedge p_{i-1} \wedge p_{i+1} \wedge \dots \wedge p_n \rightarrow q, \min(V, W))$

The mental state of agents contains a set of facts and rules. In our model, both facts and rules are weighted with truth-values in T_n , meaning that the fact or the rule is true at least to some degree. Rules are tuples $r = (m_r, c_r, \rho_r)$ where m_r is the premise (a set of atoms), c_r is the conclusion (an atom) and $\rho_r \in T_n$ is the truth-value of the rule. The representation consists of mapping each atom in Σ to its truth-value and the (possibly empty) set of rules that conclude it.

Definition 2 (Mental State) Let R be a set of rules, we define an agent mental state M of an agent A as a mapping: $M_A : \Sigma \rightarrow T_n \times 2^R$ where, for each $f \in \Sigma$, $M_A(f) = (\rho_f, R_f)$, being $R_f = \{(m_r, \rho_r) | (m_r, f, \rho_r) \in R\}$

The representation of an agent's mental state will evolve as deduction proceeds. We represent the initial mental state of an agent as a mapping from any atom into *unknown* and the set of rules deducing it. It means that the atoms initially have their most imprecise value—that is 0.

We consider that a proposition has a *definitive* value when there are no rules that can contribute to its *provisional* value (initially *unknown* or 0), producing a more precise one by means of applications of the parallel composition inference rule. We will use a proposition to specialise rules only when that proposition has a definitive value. This permits rules to be substituted by its specialised versions being the condition eliminated from its premise. When there are no conditions left in the premise of a rule the conclusion of the rule is generated.

To describe the specialization algorithm we describe first the specialisation of a rule. Given an atom (p, ρ_p) and a rule (m_r, c_r, ρ_r) and considering that $p \in m_r$ then the specialization of the rule with respect to that atom will be a new specialized rule $(m_r - \{p\}, c_r, \min(\rho_p, \rho_r))$, or a new atom if the rule had a single condition $(c_r, \min(\rho_p, \rho_r))$.

We extend now the description of the specialisation of a rule to that of the specialisation of a set of rules concluding the same atom p , the mental state can be expressed as $M(p) = (\rho_p, R)$. In doing so, we select in turn a rule $r \in R$ to specialise. If its specialisation, with respect to a fact (f, ρ_f) , returns a new rule r' then we substitute the rule by the specialised one in the agent's mental state representation, and the truth-value of p is not changed giving $M(p) = (V_p, R - \{r\} + \{r'\})$. If the rule is completely specialized and returns ρ_f , the rule is eliminated and a new truth-value for p is calculated by means of the parallel composition inference rule, and the new mental state would be $M(p) = (\max(V_f, \rho_f), R - \{r\})$.

To specialise a complete agent's mental state we will use each fact with definitive value in the mental state in turn to make specialization steps that possibly will generate definitive values for other atoms to be later used to specialise more the state.

2. Agents

In the Section above we have explained what will be considered to be part of the mental state of agents and the basic mechanisms of specialization: given new external information, the mental state of an agent is completely specialized in a data driven style. In this

Section we present the concept of agent considering that it is a goal driven entity. Apart from the passively information acquired by perception, agents proactively find new information that will be useful to satisfy their goals. Consider a multi-agent system with n agents $\mathcal{A}_n = \{A_1, \dots, A_n\}$. Each agent has the following structure:

Definition 3 (Agents) *A deliberative agent is a tuple $A_i = \langle M_i, G_i, I_i, O_i \rangle$ where:*

- I_i is the input interface, the set of external facts that can be obtained querying other agents. They are tuples $\langle x, A_j \rangle$, where $x \in \Sigma$, $A_j \in \mathcal{A}$ and $A_j \neq A_i$.
- O_i is the output interface, this is, the set of facts that the agent can answer to other agents.
- G_i are the set of goals of A_i . They are tuples $\langle x, A_j \rangle$, where $x \in \Sigma$ and $A_j \in \mathcal{A}$.
- M_i is the mental state of agent A_i .

We can see that an agent has two important elements: the mental state that is considered to be its building block, and a set of goals that guide its behavior. Goals are facts that the agent want to solve because it has commitments with other agents—generated from communication—or self commitments—internal facts not related with other agents. The input and output interface define the relation with the external world.

Definition 4 (Fact privacy) *The mental state of an agent A_i contains two kinds of facts:*

- A fact $f \in O_i$ is called public then it can be answered to other agents.
- The facts $f \notin O_i$ are called private, then they can not be revealed to any other agent.

Definition 5 (Fact state) *The mental state of an agent A_i contains three kinds of facts:*

- The facts $f \in \{p \in \Sigma | M(p) = (V_p, \emptyset), V_p \neq 0\}$ are called definitive or totally specialized because there is no more knowledge that could increase their precision.
- The facts $f \in \{p \in \Sigma | M(p) = (V_p, R), V_p \neq 0, R \neq \emptyset\}$ are called provisional or partially specialized and can be improved if there is enough information.
- The facts $f \in \{p \in \Sigma | M(p) = (0, R)\}$ are called pending and they are (provisionally) unknown.

2.1. Agents mental state cycle

When an agent's life begins and it receives a simple query, the agent can accept or reject it depending of multiple circumstances, for instance, privacy. In the case that the query is accepted, the agent begins a goal-driven—backward chaining style—work done over its mental state. This task will produce new goals (internal and external) that has to be solved. When new facts are known it is started a data-driven task of specialization—forward chaining style.

Agents can send and receive rules as conditional answers or knowledge communication. When the state of a query is *pending* or *provisional* we have to decide how to build a conditional answer. In the case of pending facts the conditional answer will be a set of rules useful to obtain a value for that fact; in the case of provisional facts the answer will be the provisional value and a set of rules useful to improve its value. When an agent receives a conditional answer it adds the new knowledge to its mental state.

Initially $G_i = \emptyset$ and all the facts have value *unknown* (0). We can summarize goal-driven work in the following steps:

1. When A_i receives a query q from an agent, and $q \in O_i$, then $G_i := G_i \cup \{ \langle q, A_j \rangle \}$
2. For each goal $\langle g, A_k \rangle \in G_i$,
 - (a) if $A_k \neq A_i$ we generate a query g to the agent A_k .
 - (b) if $A_k = A_i$ it means that the goal is a self commitment and the agent starts a search process in order to find which is the information it needs.
3. Multiple specialization steps drives to reach goals. Given a goal $\langle g, A_i \rangle \in G_i$
 - (a) If $M_i(g) = (V_g, \emptyset)$ and $V_g \neq 0$ then the agent generates a message for agent A_k with the contents (g, V_g, \emptyset) .
 - (b) If $M_i(g) = (V_g, R)$ and $R \neq \emptyset$ and $\forall (m_r, c_r, \rho_r) \in R, m_r \subseteq O_i$ then the agent generates a message for agent A_k with (g, V_g, R) .

In both cases $G_i := G_i - \{ \langle g, A_k \rangle \}$
4. When the agent receives answers from other agents, these are used to specialize the mental state. When the answer is (g, V'_g, R') and $M_i(g) = (V_g, R)$ then $M'_i(g) = (max(V_g, V'_g), R \cup R')$

The contents of answer messages are definitive facts or provisional facts with all the necessary rules to make it definitive. This does not mean that a fact with a provisional value will stop being a goal. This only means that a more precise value is reached. Stop criterion will be based on (i) goal value is found, (ii) goal is canceled or (iii) assigned time to find the goal is over (assigned time will depend on query priority and on priority agent AG wants to give it). Different criterions to choose a rule or an atom are out of the scope of this paper, in a backward chaining style we will choose the rule with best truth-value and the first premise in order of writing.

3. Communication

The communication is essential between agents because it is the base of important activities such us: cooperation, coordination and negotiation. It lets to send and receive knowledge, resolve conflicts in the tasks resolution or synchronize actions [19]. In our case, communication is the base in the conversational process between agents. Communication process is based on two important actions, these are: *querying* and *answering*.

After receiving a query, agents elaborate an answer with the information they have or get from other agents. Unquestionably the wished answer is the most precise fact value, nevertheless taking into account that there exist private facts or that their definitive values are not found yet, agents could answer with rules. Messages including rules could also be an option agents take when they have rules with facts that belong to other ones and do not want to obtain this information by themselves.

For querying or answering, agents use messages. To give a semantic to these messages, we use speech act theory [2,9] in form of performative verbs, which correspond to different types of speech acts. Based on FIPA standard [10], a message is a tuple $C_i = \langle P, S, H, B \rangle$, where P is the performative that indicates the message type (we use QUERY, ACCEPT, INFORM, REJECT and CANCEL), S (sender) is the agent that

sends the message, H (hearer) is the agent that receives the message, and B (body) is the message content.

The body of performatives QUERY, ACCEPT, REJECT and CANCEL is the name of one fact. The performative INFORM has a more complex format because it may contain facts and rules. For this performative, the body is a set of tuples $\langle M_x, V_x \rangle$ where, x is a fact, M_x is the mental state of x and V_x indicates if the value of x is *provisional* or *definitive*. Taking the example above as reference, let's see two possibilities:

- A_j knows the definitive value of f :
 $(\text{INFORM}, A_j, A_i, \{ (([1, 1], \emptyset), \text{definitive}) \})$
- Otherwise it decides to send to A_i one or a set of rules (which must not have any private fact):
 $(\text{INFORM}, A_j, A_i, \{ ((\rho_1, \{ \{a, b\}, \rho_2) \}), \text{provisional}) \})$

A dialog is a set of coherent messages: $D = \{C_1, \dots, C_n\}$. We consider those which involve only two agents, which sequentially alternate dialogue moves. Protocols [12, 8] play a central role in agent communication to specify rules of interaction between communicating agents. In our model the following protocol will be used:

1. At the beginning $D = \emptyset$.
2. A dialog D is initiated by a *query*: $(\text{QUERY}, A_i, A_j, f)$, where $A_i \neq A_j$.
 QUERY can appear, obviously, at any moment during a dialog.
3. Depending of the A_j output interface, it can accept or reject the query of A_i :
 - If $f \notin O_j$, then $(\text{REJECT}, A_j, A_i, f)$
 - If $f \in O_j$, then $(\text{ACCEPT}, A_j, A_i, f)$
4. If agent A_j has accepted, one of these five alternatives could happen:
 - (a) A_j gives A_i the definitive value of proposition requested
 $(\text{INFORM}, A_j, A_i, \{ ((\rho_1, \emptyset), \text{definitive}) \})$
 - (b) A_j gives A_i a provisional value of proposition requested
 $(\text{INFORM}, A_j, A_i, \{ ((\rho_1, \emptyset), \text{provisional}) \})$
 - (c) A_j gives A_i one or a set of rules that help to deduce or improve the value of proposition requested
 $(\text{INFORM}, A_j, A_i, \{ ((\rho_1, R), \text{provisional}) \})$
 - (d) A_i cancels the query made to A_j $(\text{CANCEL}, A_i, A_j, f)$
 - (e) A_j could need more information to give an answer and instead of answer with a rule it decides to do all by itself.
 In this case, A_j will make all necessary queries to other agents, for example:
 $(\text{QUERY}, A_j, A_k, f)$, where $A_k \neq A_i \neq A_j$, and when it have a value it will send to A_i . This makes process go to the beginning.

It is important to notice that performatives ACCEPT and REJECT allows agents to have social commitments [6]. A social commitment is defined as a structure indicating that there is a debtor committed to an action relative to a creditor [9]. In our case, when A_j accepts, it assumes a commitment with A_i , which is reflexed in its goals list.

| Phil, Agent Leader (A_l) \Rightarrow | \Leftarrow Karl, Agent Programmer (A_p) \Rightarrow | \Leftarrow Vicky, Agent Designer (A_d) |
|--|--|---|
| adapt-game@ A_p accept-adjustments project-begins | mobile-hw-supports adjust-graphics@ A_d adapt-game guarantee-impact | screen-128x128 accept-adjustments@ A_l guarantee-impact@ A_p adjust-graphics |
| rule r1 | rule r2 | rules r3 and r4 |
| $T_5 = (\text{false (0), slightly-true (st), quite-true (qt), very-true (vt), true (1)})$ r1: (adapts-game@ $A_p \rightarrow$ project-begins,0) r2: (mobile-hardware-supports \wedge adjust-graphics@ $A_d \rightarrow$ adapt-game,0) r3: (screen-128x128 \rightarrow adjust-graphics,qt) r4: (accept-adjustments@ $A_l \wedge$ guarantee-impact@ $A_p \rightarrow$ adjust-graphics,0) | | |

Figure 1. Mobile games company example.

4. Example

Consider a very simple scenario with three agents: Phil, Karl and Vicky; project leader, programmer and graphic designer respectively of a mobile games company. A new project has to be developed and Phil wants to know if Karl can do it.

–**Phil** (1): Hi Karl, there is a new project to adapt game *Drakon* for the mobile model *WX3*. Can you adapt it?

–**Karl** (2): Hi Phil, I will see the mobile and game information and I promise you to have an answer as soon as possible.

(To answer, Karl needs to analyze mobile hardware and to talk with Vicky. He sends her an e-mail with all information about the game and the mobile and call her later. Vicky analyzes the information. She knows that if minimum screen resolution is 128x128 pixels then it is possible to adjust graphics. But, for a definitive answer she would need to talk with Karl)

–**Karl** (3): Vicky, I sent you information about a new project, do you think you can adjust those graphics for model *WX3*?

–**Vicky** (4): Hi Karl, I think it is possible. However, I need to know if you guarantee me that the game will not lost its impact in users.

–**Karl** (5): Don't worry Vicky, I assure you the game won't lose its impact. Now, can I tell Phil that we will adapt the game?

–**Vicky** (6): One more thing Karl, I need Phil's agreement to make the adjusts you are suggesting. (Karl decides to talk directly with Phil about it)

–**Karl** (7): Phil, I had to talk with Vicky because if she makes some graphic adjusts I will be able to adapt *Drakon*. She said that if you agree with those adjusts, she will make them.

(At this point, Phil has all the information to know if *Drakon* can be adapted or not)

In Figure1 we can see the set of fact and rules of the agents. Now, let's see their initial state:

$$A_l \left\{ \begin{array}{l} I_l = \{(\text{adapt-game}, A_p)\} \\ O_l = \{\text{project-begins}\} \\ G_l = \{\text{project-begins}\} \\ M_l(\text{adapt-game}) = (0, \emptyset) \\ M_l(\text{accept-adjustments}) = (0, \emptyset) \\ M_l(\text{project-begins}) = (0, \{\{\text{adapt-game}\}, 1\}) \end{array} \right.$$

$$\begin{aligned}
 A_p & \begin{cases} I_p = \{(\text{adjust-graphics}, A_d)\} \\ O_p = \{\text{adapt-game}, \text{guarantee-impact}\} \\ G_p = \emptyset \\ M_p(\text{mobile-hw-supports}) = (0, \emptyset) \\ M_p(\text{adjust-graphics}) = (0, \emptyset) \\ M_p(\text{guarantee-impact}) = (0, \emptyset) \\ M_p(\text{adapt-game}) = (0, \{(\{\text{mobile-hw-supports}, \text{adjust-graphics}\}, 1)\}) \end{cases} \\
 A_d & \begin{cases} I_d = \{(\text{accept-adjustments}, A_l), (\text{guarantee-impact}, A_p)\} \\ O_d = \{\text{accept-adjustments}, \text{guarantee-impact}, \text{adjust-graphics}\} \\ G_d = \emptyset \\ M_d(\text{screen-128x128}) = (0, \emptyset) \\ M_d(\text{accept-adjustments}) = (0, \emptyset) \\ M_d(\text{guarantee-impact}) = (0, \emptyset) \\ M_d(\text{adjust-graphics}) = (0, \{(\{\text{screen-128x128}\}, qt), (\{\text{accept-adjustments}, \text{guarantee-impact}\}, 1)\}) \end{cases}
 \end{aligned}$$

(1) A_l has the objective to begin a new project. According to rule $r1$, A_l depends on A_p , therefore it sends a query: $(\text{QUERY}, A_l, A_p, \text{adapt-game})$

(2) A_p can accept or reject it, let's suppose in this example that all agents will always accept queries, then it sends: $(\text{ACCEPT}, A_p, A_l, \text{adapt-game})$ and adds a new goal to its G_p list. To achieve this goal, A_p needs to know if the game can be programmed for that mobile model (this depends on mobile hardware and A_p gets this information by itself, reading the mobile guide, and assigns a value of vt). When A_p gets this value, it proceeds to specialize. Now the mental state of A_p is:

$$A'_p \begin{cases} I_p = \{(\text{adjust-graphics}, A_d)\} \\ O_p = \{\text{adapt-game}, \text{guarantee-impact}\} \\ G_p = \{\text{adapt-game}\} \\ M_p(\text{mobile-hw-supports}) = (vt, \emptyset) \\ M_p(\text{adjust-graphics}) = (0, \emptyset) \\ M_p(\text{guarantee-impact}) = (0, \emptyset) \\ M_p(\text{adapt-game}) = (0, \{(\{\text{adjust-graphics}\}, vt)\}) \end{cases}$$

(3) The value of the rule remains very high, then it is possible to adapt the game but A_p needs to know if A_d can adjust the graphics. A_d in turn will query A_p and A_l .

(4 & 5) A_d has two rules to get adjust-graphics value, one of them only needs own information and the other one needs information from other agents. Consider there is no problem with the screen resolution. According to the original conversation, Vicky talks with Karl about game impact: $(\text{QUERY}, A_d, A_p, \text{guarantee-impact})$, and the answer is: $(\text{INFORM}, A_p, A_d, \{((1, \emptyset), \text{definitive})\})$.

$$A'_d \begin{cases} I_d = \{(\text{accept-adjustments}, A_l), (\text{guarantee-impact}, A_p)\} \\ O_d = \{\text{accept-adjustments}, \text{guarantee-impact}, \text{adjust-graphics}\} \\ G_d = \{\text{adjust-graphics}\} \\ M_d(\text{screen-128x128}) = (1, \emptyset) \\ M_d(\text{accept-adjustments}) = (0, \emptyset) \\ M_d(\text{guarantee-impact}) = (vt, \emptyset) \\ M_d(\text{adjust-graphics}) = (qt, \{(\{\text{accept-adjustments}\}, vt)\}) \end{cases}$$

(6 & 7) It is interesting to consider the meaning of the current mental state of A_d : *with the current information adjust-graphics is quite true, but if Phil considers that accept-adjustments were true then adjust-graphics will be very true*. A_d needs one more value from A_l . It can ask A_l , but it decides to pass the job to A_p , and sends this new rule: $(\text{INFORM}, A_d, A_p, \{((\text{adjust-graphics}; p; \{(\text{accept-adjustments}, vt)\}), \text{provisional})\})$. A_p can do nothing with this rule; it could ask to A_l about accept-adjustments but this is not an exportable

fact, then A_l can not give any answer. So that, A_p sends its own rule together with A_d rule.

$$A_p'' \left\{ \begin{array}{l} I_p = \{(\text{adjust-graphics}, A_d), (\text{accept-adjustments}, A_l)\} \\ O_p = \{\text{adapt-game}, \text{guarantee-impact}\} \\ G_p = \{\text{adapt-game}\} \\ M_p(\text{mobile-hw-supports}) = (qt, \emptyset) \\ M_d(\text{adjust-graphics}) = (qt, \{(\text{accept-adjustments}, vt)\}) \\ M_p(\text{guarantee-impact}) = (0, \emptyset) \\ M_p(\text{adapt-game}) = (0, \{(\text{adjust-graphics}, vt)\}) \end{array} \right.$$

(8) A_l has now all the necessary information to say if *project-begins* is *quite true* or *true*. Depending on the value of *finaldecision* it will be *qt*—when *finaldecision* is *false*—or *vt*—when it is *true*.

$$A_l' \left\{ \begin{array}{l} I_l = \{\text{adapt-game}, A_p\} \\ O_l = \{\text{project-begins}\} \\ G_l = \{\text{project-begins}\} \\ M_d(\text{adjust-graphics}) = (qt, \{(\text{accept-adjustments}, vt)\}) \\ M_p(\text{adapt-game}) = (0, \{(\text{adjust-graphics}, vt)\}) \\ M_l(\text{accept-adjustments}) = (\text{finaldecision}, \emptyset) \\ M_l(\text{project-begins}) = (0, \{(\text{adapt-game}, 1)\}) \end{array} \right.$$

5. Conclusions

In this paper we have presented how the specialization of rule-based knowledge bases can be the central mechanism to deliberate and also to produce *reasonable* dialogs among conversational agents [18,3]. Agents communicate exchanging data and knowledge in the form of conditional answers to solve their goals in a collaborative manner. The contents of the messages can be part of the mental state of agents, containing only public information. We believe that this model makes sense when we manage imperfect information: vague, imprecise and incomplete. In this case the specialization mechanism give new opportunities of richer conversations by using in each moment the more precise information to drive the questioning/answering protocols.

One important point not covered in this paper is related to the use of negation in the conclusions of rules. In our complete language a fact a has the value $[\alpha, \beta]$ because rules concluding a are responsible of α (the minimum of the interval) and rules concluding $\neg a$ of β (the maximum). More certain rules produces more precision for the conclusion. Provisional values for facts are those less precise that can be used also to produce provisional specialization and so provisional values for other facts.

Another important issue is time. It may be reasonable to think in different strategies of specialization using provisional values, i.e. when a concrete timeout has been reached or when we need a value, we can use a less precise but useful result, similar to *anytime* algorithms. The pass of time gives an opportunity to increase the accuracy, then the goals of agents can persist until it is no possible to obtain more precise values.

What we need to do now is to carry out experiments to see which are the emergent conversations among agents; to study different strategies for obtaining information: in parallel, using provisional values, etc.; to study different kind of collaborative effort and delegation [5] and coordination [4]; and to extend our model by adding concepts related to the Electronic Institution model [1].

Acknowledgements.

Authors acknowledge partial support by the Spanish projects IEA (TIN2006-15662-C02-01), MULO2 (TIN2007-68005-C04-01) and Agreement Technologies (CON-SOLIDER CSD2007-0022, INGENIO 2010). Mariela Morveli-Espinoza is supported by the Programme Alβan, the European Union Programme of High Level Scholarships for Latin America, scholarship No.(E06D101440PE). We would like to thank the referees for their valuable suggestions and comments.

References

- [1] Josep Lluís Arcos, Marc Esteva, Pablo Noriega, Juan A. Rodríguez-Aguilar, and Carles Sierra. Engineering open environments with electronic institutions. *Engineering applications of artificial intelligence*, 18(2):191–204, 2005.
- [2] J.L. Austin. *How to Do Things with Words*. Harvard University Press, 1962.
- [3] M Barbuceanu and WK Lo. Conversation oriented programming for agent interaction. In *Issues in Agent Communication*, volume 1916 of *Lecture Notes in Artificial Intelligence*, pages 220–234, 2000.
- [4] J Bentahar, B Moulin, and B Chaib-draa. Commitment and argument network: A new formalism for agent communication. In *Advances in Agent Communication*, volume 2922 of *Lecture Notes in Artificial Intelligence*, pages 146–165, 2003.
- [5] SE Brennan and Hulteen. Interaction and Feedback in a Spoken Language System - A Theoretical Framework. *Knowledge-Based Systems*, 8(2–3):143–151, APR-JUN 1995.
- [6] M. A Colombetti. Commitment-based approach to agent speech acts and conversations. In *Fourth International Conference on Autonomous Agents, Workshop on Agent Languages and Conversation Policies*, pages 21–29, 2000.
- [7] Robert Demolombe. A strategy for the computation of conditional answers. In *Proceedings ECAI’92*, pages 134–138, 1992.
- [8] U. Endriss, N. Maudet, F. Sadri, and F. Toni. Protocol conformance for logic-based agents. In *Proceedings IJCAI-2003.*, 2003.
- [9] Roberto A. Flores and Rob Kremer. Bringing coherence to agent conversations. In *AOSE*, pages 50–67, 2001.
- [10] Foundation for Intelligent Physical Agents (FIPA). *Communicative Act Library Specification*, 2002.
- [11] T Fruhwirth. Specialization of concurrent guarded multi-set transformation rules. In *Logic Based Program Synthesis and Transformation*, volume 3573 of *Lecture Notes in Artificial Intelligence*, pages 133–148, 2005.
- [12] Mark Greaves, Heather Holmback, and Jeffrey Bradshaw. What is a conversation policy? In *Issues in Agent Communication*, pages 118–131. Springer-Verlag: Heidelberg, Germany, 2000.
- [13] K Iwanuma and K Inoue. Minimal answer computation and SOL. In *Logics in Artificial Intelligence 8th*, volume 2424 of *Lecture Notes in Artificial Intelligence*, pages 245–258, 2002.
- [14] J. W. Lloyd and J. C. Shepherson. Partial evaluation in logic programming. *The Journal of Logic Programming*, 11(3/4):217–242, October/November 1991.
- [15] J. Puyol, L. Godo, and C. Sierra. A specialisation calculus to improve expert system communication. In Bern Neumann, editor, *Proceedings of the 10th European Conference on Artificial Intelligence, ECAI’92*, pages 144–148, Vienna, 1992. John Wiley & Sons, New York.
- [16] J. Puyol-Gruart, L. Godo, and C. Sierra. Specialisation calculus and communication. *International Journal of Approximate Reasoning (IJAR)*, 18(1/2):107–130, 1998.
- [17] J. Puyol-Gruart and C. Sierra. Milord II: a language description. *Mathware and Soft Computing*, 4(3):299–338, 1997.
- [18] F Rago. Conversational agent model in intelligent user interface. In *Fuzzy Logic and Applications*, volume 2955 of *Lecture Notes in Artificial Intelligence*, pages 46–54, 2006.
- [19] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Englewood Cliffs, NJ, 1995.
- [20] Yoav Shoham. Agent-oriented programming. *Artificial Intelligence*, 60:51–92, 1993.

Robustness in Recurrent Auctions for Resource Allocation

Victor MUÑOZ and Dídac BUSQUETS

{vmunozs,busquets}@eia.udg.es

University of Girona

Abstract. Resource allocation problems where resources have to be assigned to tasks in such a way that no resource gets overused can be solved using recurrent auctions. In dynamic environments where unexpected changes may occur, searching the optimal solution may not be the best choice as it would be more likely to fail. In these cases a robust solution is preferable. In this paper we present a robustness mechanism for auctions, producing feasible and near optimal solutions even if non-planned events occur.

Keywords. Robustness, auctions, resource allocation

1. Introduction

Many real-world applications pose the problem of *resource allocation*, such as assigning memory and computing power to processes, distributing tasks to machines in a factory, or selecting the personnel to carry out certain jobs. This is an optimization problem where a set of resources is assigned to a set of agents (the entities needing the resources). The goal is to find a solution that maximizes or minimizes a given objective function (such as cost, revenue, makespan, etc.), while fulfilling a set of constraints (usually regarding the resources). However, finding the optimal solution is not always the best choice, since it could fail in case the environment changed (a machine breaking down, a process taking longer than expected, etc.). Therefore, it would be desirable to have a *robust solution* that could still be applicable even if unexpected events occurred. Obviously, the price of robustness is optimality [2], since usually a robust solution will be suboptimal. Therefore, there is a need of balancing the tradeoff between optimality and robustness.

In some scenarios where the resources can only be used temporally, the allocation process is continuously repeated. Moreover, in real environments (such as industrial ones), the access to resources is vital to perform tasks. Thus, it could be the case that an agent uses more resources than those it was allocated, or even use some resource without having been authorized to do so. The consequence of such behavior would be resource overuse, which would negatively affect the rest of the agents. One way of addressing this problem is by imposing fines or penalties to those agents not obeying the allocation [13]. However, even with that, an agent may still prefer to pay a penalty if it obtains a better profit by using the resources when not authorized. Therefore, in order to prevent such a situation, the allocation should incorporate some degree of robustness. Then, even if some agents disobeyed, the probability of having resource overuse would be lowered.

In this paper we focus on market-based mechanisms, namely auctions, for assigning resources to agents. Auction mechanisms have become a popular approach for dealing with resource allocation problems [4]. One of the advantages is that they provide privacy to the agents, since they do not need to disclose too much private information, and they also provide more autonomy in the decision-making, in comparison with purely centralized approaches. However, the problem of deciding the winners of an auction, known as the *Winner Determination Problem*, is a complex optimization problem, and most of the developed algorithms for solving it are focused on finding optimal solutions instead of robust ones.

Although robustness has already been studied in the field of mechanism design, it has been usually to tackle the problem of false-name bids [14] or to achieve mechanisms that are strategy-proof (that is, the agents' best strategy is truthful bidding) [5]. However, as mentioned before, we are dealing with robustness at the solution level, that is, solutions that are still valid even if the conditions for which they were computed change. This kind of robustness has been addressed in the planning and scheduling field [1,6,3], but, as far as we know, the only approach that deals with such robustness in auctions has been presented in [8]. This work uses the concept of super-solutions [7] to address the problem of bid withdrawal and generates solutions with a repair cost below a given threshold. However, it is not applicable to the disobedience problem presented above. Thus, we have focused our efforts in developing a mechanism for adding robustness to auctions for scenarios with potentially disobeying agents.

2. Auction-based Resource Allocation

Resource allocation problems can be solved using classical IA techniques, usually based on centralized approaches where a central element makes all the decisions. However, in recent years auctions are being increasingly used for these problems, as they are more indicated for distributed problems where the participants do not want to disclose private information related to their internal functioning upon which their decisions are based [4].

An auction-based distributed scenario for resource allocation is composed of a central element (coordinator) representing the shared resources of certain capacities, and a set of individuals (agents) competing for the resources. Agents that want to use the resources for a given period of time send requests to the coordinator composed by the resource/s that they want to use, and the required period of time. Formally, a request is defined as $\{s_i, d_i, \bar{q}_i\}$, where s_i and d_i are, respectively, the start time and duration of the use of the resources and \bar{q}_i is the capacity requirements of the resources ($q_i = \{q_{i,1}, q_{i,2}, \dots, q_{i,n}\}$ where n is the total number of resources). The resources are then assigned to the requesters using an auction. The goal of the auction is to avoid overuses of the resources by selecting a subset of the requests, which will gain the right to use some resources for a given time period, while the remaining should wait for another opportunity. The selection criteria is based on the bids submitted by the agents.

Formally, the problem to solve in an auction where there are multiple resources of different capacities is named the Winner Determination Problem (WDP) for multi-unit combinatorial auctions [9] (similar to the multi-dimensional knapsack problem):

$$\max \sum_{i=1}^{NR} x_i \cdot v_i \quad \text{s.t.} \quad \sum_{i=1}^{NR} x_i \cdot q_{i,j} \leq Q_j \quad \forall j \in C \quad (1)$$

where NR is the number of requests, $x_i \in \{0, 1\}$ represents whether request i is denied or authorized, $v_i \in \mathbb{R}^+$ is the bid value for request i , $q_{i,j}$ is its capacity requirement for the resource j , Q_j is the resource j capacity, and C is the set of resources.

Note that in a resource allocation environment where the agents continuously need the use of the resources, the auction is repeated several times. Concretely, each time the requests overuse the resources an auction is executed to decide which of the requests to authorize. This scenario where the bidders are continuously competing for the resources is known as a *recurrent auction*, and gives rise to a new problem called the Bidder Drop Problem which happens when an agent participating in many auctions is always losing [11]. In such a case, the agent could decide to stop participating in the auction or stop obeying the outcome of the auction. This problem has been typically addressed using fairness mechanisms [10,12]. However, although fairness incentivizes agents to participate in auctions, it does not address the robustness of the system. Robustness is a desired feature in these situations, as it would produce solutions taking into account the agents which are most likely to disobey, thus preventing overuse of the resources.

3. Robustness in Auctions

As mentioned before, in some domains an interesting feature on auctions is to incorporate robustness representing the ability of a solution to overcome unexpected changes in the environment. In such situations we are willing to accept a suboptimal solution in order to ensure that it remains feasible and near optimal even when the environment changes. There are two general approaches for achieving robustness in uncertain environments:

- *Reactive robustness* addresses the problem of how to recover from a disruption once it has occurred, for example providing an alternative solution.
- *Proactive robustness* is concerned in finding a solution that takes into account the possible events in the environment and therefore, the solution is robust by itself.

While reactive robustness has been quite studied in combinatorial auctions in the work of Alan Holland [8], proactive robustness is still relatively unexplored. In the following, we will design a proactive robustness mechanism for recurrent combinatorial auctions that considers nearly every possible change on the auction. The mechanism is based on a building a model of the participants in the auction that is learned in successive clearings of the auctions. The robustness mechanism consists in three main components:

- **Trust model** of the agents requesting the resources
- **Risk function** of the agent selling the resources (the *auctioneer*, or *coordinator*)
- **Robust solution generation**

The first component (the trust model) is concerned with the agents requesting resources. It is a main part of the mechanism as it models the behavior of the agents by learning from their actions their behavior and the circumstances in which an agent is most likely to disobey the decisions of the coordinator. The second component is related to the coordinator and its risk function, as the concept of a robust solution varies depending on the risk attitude of this concrete agent. Finally, with the inputs coming from all the agents, the robustness of the system is achieved by combining the risk of the coordinator with the trust on the agents requesting the resources to generate a solution that is robust, that is, it is able to absorb some level of changes in the environment.

3.1. Trust model

An agent requesting resources to perform tasks can disobey the decisions of the auctioneer for several reasons. It is not usually the case that an agent disobeys every decision of the auctioneer independently of the characteristics of the task to perform. Normally, an agent would disobey only the decisions that deny some tasks that it needs to perform for some reason. Therefore the trust model should not contain only a unique global value for the degree of trust of an agent, but the trust value should be related to a given task features. Possible task features to build the trust model with include the resources capacity requirements, the task duration, etc.

The trust model is learned during the recurrent auction storing not only the probability of disobeying of the agents, which happens when an agent uses the resource when it is not authorized to, but also its lying magnitude, representing the difference between the requested capacity of the resources and the real used capacity, as in some scenarios an agent may request to perform some tasks using a given capacity of resources and later use a higher capacity than requested. Consequently, the measures stored by the trust model are the following:

- **Probability of disobeying.** This value $\in [0..1]$ can be measured in many different ways, being the most obvious the average of disobediences in relation to the total number of auctions the agent has been involved in. However, it could be measured counting also the times where the agent has performed the authorized task but using a higher amount of capacity than requested.
- **Lie magnitude.** This value $\in [0..\infty]$ represents the degree of the disobedience. For example a value of 1 would represent that when the agent disobeys, it uses the quantity of resources requested for the task, while a value of 1.5 would represent that it uses 150% of the requested capacity.

A graphical representation of this trust model using only one characteristic of the task is shown in Figure 1 (to use more task characteristics, additional dimensions would be added). Note that this model is general enough to allow including even the case where an industry does never disobey the auctioneer, but it uses a higher amount of capacity than requested (having a lie magnitude greater than 1 at disobey probability of 0). This is particularly useful in problems where the resource capacity requirements of the agents are quite dynamic.

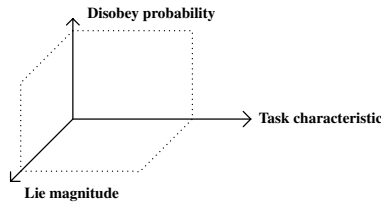


Figure 1. Trust model.

The trust model is learned by the auctioneer agent at execution time. Every time a task is performed the trust model of the respective agent is updated checking firstly if the task has been performed after the authorization of the auctioneer or not, that is, the agent has disobeyed the result of the coordination (the solution of the auction), and secondly if the resource capacity used is the same as what was requested.

3.2. Risk function

The risk attitude of the auctioneer characterizes the tradeoff between robustness and optimality that it wants, given that robustness and optimality are contradictory objectives. The risk function of the coordinator can be also seen as its willingness to face dangerous situations.

Risk attitudes are generally categorized in three distinct classes: risk averse, neutral and proclive. Risk aversion is a conservative attitude for individuals who do not want to be at stake. Risk neutral agents display an objective predilection for risk, whilst agents with a proclivity for risk are willing to engage in situations with a low probability of success. For example, a risk-averse auctioneer would consider that every request with a probability of disobeying greater than 0 is going to use the resources even if unauthorized, and thus it would auction only the remaining resources capacities over the rest of the requests. On the other hand a risk-proclive auctioneer would consider that if a request has a low probability of being disobeyed, it would not be the case at this time and hence the auctioneer would auction a bigger amount of resources capacities, although with a higher risk of being overused.

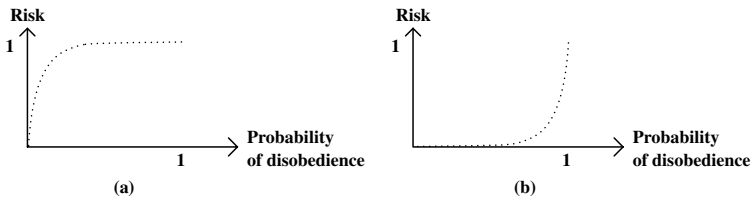


Figure 2. Risk attitude function: (a) averse, (b) proclive.

The risk function f_{risk} defines the risk attitude of the auctioneer (between 0 and 1) as a function of the probability of disobeying of a given agent and a given request. An example of a risk function is shown in Figure 2(a). In this case it represents a risk-averse auctioneer, since the resulting *risk value* is almost always 1 (it considers risky requests as if they are going to surely use the resources even if unauthorized), regardless of the probability of disobeying. On the other hand, a risk-proclive auctioneer would have the final value almost always set to 0, as seen in Figure 2(b), and a risk-neutral one would have it set accordingly to the probability of disobeying.

3.3. Robust solution generation

The trust model and the risk function of the coordinator are used to generate the *robustness constraint* that will provide robust solutions according to the proactive approach. This constraint is added to the constraint optimization problem (previously formulated in Equation 1) related to the auction, in order to force the solution to be robust.

In the auction (executed each time a conflict is detected) the auctioneer is faced with a set of requests (the tasks involved in the conflict), each with trust features associated obtained from the trust model. Then the auctioneer decides which requests to authorize in function of its risk attitude.

The robustness constraint is formulated in a way that the solution finds a balance between the amount of resources required by the authorized requests and the assumed risk

from the unauthorized requests (appropriately weighted by its probability of disobeying, lie magnitude and the risk function f_{risk} of the auctioneer). The objective is not to exceed the maximum capacities of the resources (Q_j). This constraint is defined as shown in Equation 2.

$$\sum_{i \in [1, n]} x_i \cdot c_i + \sum_{i \in [1, n]} (1 - x_i) \cdot c_i \cdot f_{risk}(P_i) \cdot M_i \leq Q_j \quad \forall j \in C \quad (2)$$

The first summatory represents the resources used by the authorized requests, while the second summatory characterizes the resources potentially used by the unauthorized requests. Hence, the unauthorized requests are considered as if they were performed in the cases where the probability of disobeying of the associated agent (P_i) is higher than zero. However this value (appropriately weighted with its corresponding lie magnitude M_i) is considered as a function of the risk attitude of the auctioneer f_{risk} . In this case we have considered that the lie magnitude is directly multiplied by the *risk value*, but another function could be used as well.

Another way of understanding this equation is by moving the second summatory to the right side. Then it can be read as if a concrete capacity of the resource/s is reserved to be used by the unauthorized tasks that are likely to be disobeyed and performed anyway.

4. Experimentation

To test the robustness mechanism previously described, we have used a real-world problem: the Waste Water Treatment Plant Problem (WWTPP). The main components in this problem are the Waste Water Treatment Plant and the set of industries performing waste discharges to the sewage. The job of the treatment plant is to process the sewage coming from the industries, removing its contaminants in order to return a clean waterstream back to the river. If the discharges are done without any coordination, the amount of water arriving at the plant may exceed its capacity, which causes the overflow to go directly to the river without being treated and increasing its contamination. Thus, in order to prevent such dangerous environmental situation, the industrial discharges should be temporally distributed so that all of them can be fully treated.

We assume that each industry has a retention tank (of a given capacity) where it can store a discharge whenever it is not authorized, and empty it later on. In this scenario the recurrent auction mechanism will determine which discharges to authorize and which to be temporarily stored in the tank in order to not exceed the plant's capacity.

Regarding the robustness mechanism proposed in this paper, it is easier to understand more clearly with this concrete problem why it is useful. In this scenario it is conceivable that industries may sometimes disobey the decisions of the plant. The most obvious reason is when an industry has its retention tank completely full; in this case if the forthcoming discharge is not authorized, the industry will be forced to discharge it anyway, thus disobeying the plant. However, an industry could disobey the decisions of the plant for other uncontrolled and unpredictable reasons, for example when an industry cannot use its retention tank (for maintenance purposes, for instance), or when a concrete discharge cannot be stored in the tank because of its high level of contamination, etc. That is the reason why the robustness mechanism has been designed considering the characteristics of the task in the trust model.

4.1. Solving the WWTPP

The WWTPP can be modeled as a recurrent combinatorial auction, where the auctioneer is the treatment plant, the resource being sold is its capacity, and the agents using the resource are the industries that perform discharges. Here the resource consumption (as well as the individual discharges) does not have only a global capacity limit (hydraulic capacity), but it is extended with many thresholds, one for each contaminant type. The goal of the auctioneer is not to exceed any of its thresholds (hydraulic capacity and contaminant levels).

The coordinating scenario described in the previous sections can be easily adapted to be applied to this problem, so the robustness mechanism can also be used.

4.2. Implementation

To evaluate the coordination and robustness mechanisms we have implemented a prototype of the system reproducing the coordination and communication process between plant and industries. So far we have only considered the hydraulic capacity of the plant. Industry agents calculate their bids taking into account the urgency to perform a discharge, based on the percentage of occupation of the tank. In case an industry agent is denied to perform one of its discharges, it first tries to store the rejected discharge into the tank, scheduling the discharge of the tank as its first activity after the current conflict finishes. If the industry has its tank already full, the discharge is performed anyway.

The free linear programming kit GLPK (GNU Linear Programming Kit) has been used to solve the winner determination problem related to each (multi-unit) auction, modeling it as a mixed integer programming problem. The robustness constraint is added as an additional constraint.

The trust models of the industries have been implemented using only one characteristic of the discharges: the flow. The models of the industries are learned during the execution by storing the total number of lies and truths (that is, disobedient and obedient actions), together with a value to compute the lie magnitude. These values are updated after each performed discharge in the following way: if the industry was authorized then the number of truths of the corresponding flow is incremented; otherwise the number of lies is incremented. Independently, the lie magnitude is computed as the difference between the used capacity and the requested capacity.

4.3. Experimentation results

Results have been evaluated considering some quality measures based on different characteristics of the solution:

- **number of overflows (NO)** occurred during the simulation
- **maximum flow overflowed (MFO)**, measured in m^3/day
- **total volume overflowed (VO)**, in liters
- percentage of discharge denials **obeyed** by the industries (**%IO**)

The experiments consisted of simulations using a set of real data provided by the Laboratory of Chemical and Environmental Engineering (LEQUIA). This data is composed of the discharges of 5 industries in two weeks. The first one is a pharmaceutical industry; it is increasing its discharge flow during the week and does not discharge during

the weekend. The second one is a slaughterhouse that discharges a constant flow, except at the end of the day when it increases. The third one is a paper industry that discharges a constant flow during the seven days of the week. The fourth one is a textile industry, whose discharges flow oscillates during the day. The fifth one is the waste water coming from the city, whose flow is fixed. The hydraulic capacity of the plant is 32000 m³/day.

We have tested the mechanism in different scenarios and situations. In the first scenario there is no coordination among the industries (without coordination the industries perform its initial discharges plans, and the treatment plant does never unauthorise any discharge). The second scenario uses the coordination mechanism and assumes that the industries always obey the decisions of the plant, as long as they have enough tank capacity. In the third scenario we introduce a probability of disobeying the outcome of the coordination mechanism. This probability depends on the occupation of the tank (the higher the occupation, the higher the chances of disobeying); a graphical representation of this function is shown in Figure 3. Two variations of the disobeying probability of disobeying have been tested.

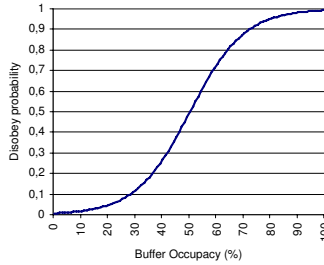


Figure 3. Disobey probability function.

Additionally, we have tested the system with another scenario where there is one single industry (the textile, chosen randomly), that will always disobey the decisions of the plant if any of its discharges is unauthorized. Every scenario has been tested activating and deactivating the robustness mechanism and with different risk attitudes of the coordinator (averse, neutral and proclive).

The outcome of all the scenarios is shown in Table 1, with the average and deviation (in brackets) of 10 simulations performed for each scenario. Concretely, we can notice that the non-coordinating scenario produces the worst results regarding volume overflowed (which is the most important indicator), while the auction-based system improves the results, principally when all the industries obey (this reflects the best possible circumstances). With disobeying industries we can notice a subtle improvement when using the robustness mechanism in both the volume and maximum flow overflowed yet the difference is not much relevant, and the number of overflows is generally higher. Regarding the risk attitude of the coordinator we do not observe its effects in this scenario.

In the environment where there is one industry always disobeying, the robustness mechanism seems to mark differences given that all the indicators are significantly improved, specially regarding the volume overflowed and percentage of obedience. However, in this scenario, like in the previous, the different risk attitudes do not produce clear differences in the outcome.

| | | | | NO | MFO | VO | %IO |
|------------------------------------|---------------|----------|--|------------------|--------------------|----------------------------------|-----------------|
| No coordination | | | | 80 | 9826 | 15.21·10 ⁶ | - |
| Obey | | | | 28 | 4996 | 3.74·10 ⁶ | 98.95 |
| Low Disobedience | No Robustness | | | 77.60 (4.12) | 14432 (865.93) | 11.5·10 ⁶ (216866) | 98.55 (0.12) |
| | Robustness | Averse | | 78.70 (7.15) | 14360 (1522) | 11.3·10 ⁶ (261362) | 98.27 (1.57) |
| | | Neutral | | 79 (7.83) | 13531 (1396) | 11.4·10 ⁶ (260669) | 98.19 (0.24) |
| | | Proclive | | 84.1 (5.16) | 14052 (1006) | 11.3·10 ⁶ (251712) | 98.15 (0.17) |
| Medium Disobedience | No Robustness | | | 126.60 (6.13) | 14398 (1604) | 13.3·10 ⁶ (363484) | 96.48 (0.31) |
| | Robustness | Averse | | 126.60 (6.13) | 14398 (1604) | 13.3·10 ⁶ (363484) | 96.48 (0.31) |
| | | Neutral | | 122.9 (6.84) | 13966 (803) | 13.2·10 ⁶ (403934) | 96.61 (0.32) |
| | | Proclive | | 121.3 (7.94) | 14233 (1358) | 13.2·10 ⁶ (374673) | 96.58 (0.41) |
| TEXTILE INDUSTRY ALWAYS DISOBEYING | | | | | | | |
| No coordination | | | | 80 | 9826 | 15.21·10 ⁶ | - |
| Obey | No Robustness | | | 112 | 6523 | 6.89·10 ⁶ | 90.84 |
| | Robustness | | | 58 | 6590 | 5.47·10 ⁶ | 96.77 |
| Low Disobedience | No Robustness | | | 112 (6.09) | 14955 (1201.58) | 12.6·10 ⁶ (233076) | 90.98 (0.2) |
| | Robustness | Averse | | 77.70 (3.68) | 14225 (1212) | 11.8·10 ⁶ (205150) | 96.69 (1.57) |
| | | Neutral | | 82.5 (7.66) | 15110 (997) | 11.9·10 ⁶ (199074) | 96.66 (0.16) |
| | | Proclive | | 81.2 (4.44) | 14018 (1596) | 11.8·10 ⁶ (133988) | 96.68 (0.18) |
| Medium Disobedience | No Robustness | | | 119.70 (4.72) | 14819 (1373.74) | 14.3·10 ⁶ (263955) | 89.96 (0.28) |
| | Robustness | Averse | | 109.50 (3.95) | 14150 (1310) | 13.6·10 ⁶ (242619) | 95.19 (0.17) |
| | | Neutral | | 113.5 (5.5) | 13708 (1040) | 13.6·10 ⁶ (445501) | 95.16 (0.37) |
| | | Proclive | | 110.9 (8.16) | 14522 (1571) | 13.6·10 ⁶ (338985) | 95.31 (0.29) |

Table 1. Simulation results.

5. Conclusions and Future Work

In this paper we have presented a proactive robustness mechanism for auction-based resource allocation problems. Through this mechanism, the system finds a solution that is robust, i.e. it is able to remain applicable even with changes in the environment. Changes involve both modifications on the resources capacities requests and using the resource

when the user is not authorized to. The core of the robustness mechanism consists in a trust model that is learned during the execution and a risk function associated to the auctioneer of the resources, that are used together in order to produce a robust allocation.

Results obtained through simulation using real data show that the robustness mechanism improves the results over the non-robust approach. However, further work has to be made in the risk attitudes of the auctioneer as we have not noticed significant changes when varying it. Also the trust model needs to be improved, as it considers tasks with different characteristics independently, yet in problems where the tasks characteristics were too dynamic it would be useless as there would not be two identical tasks.

It should be noted that the robustness mechanism may induce the agents to disobey. Different mechanisms to avoid this situation have already been studied, as for example the addition of *finés* (or *penalties*) to be paid whenever an agent does not obey; another method would be to stipulate a deposit to be paid for the participants before beginning the coordination, and returned later only to the obeying agents. However, the price of these fines or deposits should be studied in more detail in order to make it not too cheap so an agent would prefer to pay it instead of obeying the coordinator, neither too expensive so that a poor agent would have more problems than a rich one to pay it.

References

- [1] A. Ben-Tal and A. Nemirovski, 'Robust solutions of uncertain linear programs', *Operations Research Letters*, **25**, 1–13, (1999).
- [2] D. Bertsimas and M. Sim, 'The price of robustness', *Operations Research*, **52**(1), 35–53, (2004).
- [3] J. Branke and D.C. Mattfeld, 'Anticipation and flexibility in dynamic scheduling', *International Journal of Production Research*, **43**(15), 3103–3129, (2005).
- [4] Y. Chevaleyre, P.E. Dunne, U. Endriss, J. Lang, M. Lemaître, N. Maudet, J. Padget, S. Phelps, J.A. Rodríguez, and P. Sousa, 'Issues in multiagent resource allocation', *Informatica*, **30**, 3–31, (2006).
- [5] R.K. Dash, P. Vytelingum, A. Rogers, E. David and N.R. Jennings, 'Market-Based Task Allocation Mechanisms for Limited-Capacity Suppliers', *IEEE Transactions on Systems, MAN, and Cybernetics-Part A: Systems and Humans*, **37**(3), 391–405, (2007).
- [6] A.J. Davenport, C. Gefflot, and J.C. Beck, 'Slack-based techniques for robust schedules', in *Proceedings of the Sixth European Conference on Planning (ECP-2001)*, pp. 7–18, (2001).
- [7] E. Hebrard, B. Hnich, and T. Walsh, 'Super solutions in constraint programming', in *Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems*, Lecture Notes in Computer Science, 157–172, Springer, (2004).
- [8] A. Holland and B. O'Sullivan, 'Truthful risk-managed combinatorial auctions', in *Proceedings of IJ-CAI'07*, pp. 1315–1320, (2007).
- [9] J. Kalagnanam and D. Parkes, *Handbook of Supply Chain Analysis in the E-Business Era*, chapter Auctions, bidding, and exchange design, Kluwer Academic Publishers, (2005).
- [10] J.S. Lee and B.K. Szymanki, 'A novel auction mechanism for selling time-sensitive e-services', *Proc. 7th International IEEE Conference on E-Commerce Technology (CEC'05)*, pp. 75–82., (2005).
- [11] J.S. Lee and B.K. Szymanki, 'Auctions as a Dynamic Pricing Mechanism for e-Services', *Service Enterprise Integration*, pp. 131–156., (2006).
- [12] V. Muñoz, J. Murillo, D. Busquets, and B. López, 'Improving water quality by coordinating industries schedules and treatment plants', in *Proceedings of the Workshop on Coordinating Agents' Plans and Schedules (CAPS)*, ed., Mathijs Michiel de Weerd, pp. 1–8. IFAAMAS, (2007).
- [13] T. Sandholm and V. Lesser, 'Leveled commitment contracting: A backtracking instrument for multiagent systems', *AI Magazine*, **23**(3), 89–100, (2002).
- [14] M. Yokoo, Y. Sakurai and S. Matsuura, 'Robust Combinatorial Auction Protocol against False-name Bids', *Artificial Intelligence Journal*, **130**(2), 167–181, (2001).

Using Electronic Institutions for Hospitals Chronic Disease Management and Purchasing System

Ashkan Musavi¹, Maite Lopez-Sanchez¹, Jordi Campos¹ and Marc Esteve²

¹WAI research group. MAiA dept. Universitat de Barcelona,
ashkanmusavi@gmail.com, {maite, jcampos}@maia.ub.es

²Institut d'Investigació en Intel·ligència Artificial (IIIA-CSIC)
marc@iiia.csic.es

Abstract. Use of multi-agent systems (MAS) in health-care domains is increasing. MAS is an appropriate technique for many medical domains due to the characteristics of the problems in this area. According to the World Health Organization (WHO) health care systems worldwide are faced with the challenge of responding to the needs of people with chronic conditions such as diabetes, heart failure and mental illness. Chronic disease management (CDM) is a systematic approach to improve health care for people with chronic disease. Electronic Institutions (EI), as agent technology, is considered a suitable paradigm for complex environments providing structured frameworks for multi-agent systems to regulate agents' interactions. In this paper we introduce the HCDMP System which is an electronic institution to improve the Hospitals Chronic Disease Management and Purchasing processes. HCDMP system provides a number of recommended services to the patients and aims to advance the self care as a well proven and highly effective means of improving the care of chronic diseases.

1 Introduction

In recent years there has been an increasing interest in integrating the organisational concepts into multi-agent systems with the purpose of considering organisation-centred designs of multi-agent systems [1]. Using multi-agent systems and lately electronic institutions in medical and health-care domains has recently amplified [2]. Electronic Institutions (EI) [3] are a system approach to implement interaction conventions for the agents who can establish commitments on an open environment. These agents can be either humans or computer software. The proposed EI in this paper for hospitals chronic disease management and purchasing system, HCDMP, is a good example of such efforts to tackle medical situations using agent technology and electronic institutions.

Chronic diseases are those that can only be controlled and not, at present, cured [4]. Living with a chronic disease has a significant impact on a person's quality of life and on their family. The frequency of such diseases increases with age. Many older people are living with more than one chronic condition and this means that they face particular challenges, both medical and social. The care of people with chronic conditions also consumes a large proportion of health and social care resources. People with chronic conditions are significantly more likely to see their GP (General Practice), accounting for about 80% of GP consultations, to be admitted as inpatients, and to use more inpatient days than those without such conditions. The World Health Organisation (WHO) has identified that such conditions will be the leading cause of disability by 2020 and that, if not successfully managed, will become the most expensive problem

for health care systems [5]. This confirms the global importance of chronic disease management (CDM).

HCDMP system aims to implement agent technology, using *Islander* tool [6], in chronic disease management and attempts to achieve better CDM through providing recommended services, by the responsible authorities, to the chronic patients.

The rest of this paper is structured in five sections. Section 2 is about chronic disease management and introduces some examples of successful CDM. In section 3 Electronic Institutions is explained followed by a full description of the HCDMP system in section 4. Section 5 presents the related work and Metastorm Manager's Edition Designer. Finally section 6 is about future work and conclusions.

2 Chronic Disease Management

Chronic disease is the biggest problem facing health care systems worldwide seems unarguable. They include diabetes, asthma, arthritis, heart failure, chronic obstructive pulmonary disease, dementia and a range of disabling neurological conditions. Chronic diseases are the ones which current medical interventions can only control not cure. The life of a person with a chronic condition is forever altered and sadly there is no return to normal. Chronic conditions have a profound effect on the physical, emotional and mental well-being of individuals, often making it difficult to carry on with daily routines and relationships. However, in many cases, deterioration in health can be minimized by good care. Chronic disease management (CDM) [7] is an approach to health care that emphasizes helping individuals maintain independence and keep as healthy as possible through prevention, early detection, and management of chronic conditions. Figure 1 shows the results of good CDM in UK, US and Canada where, first-class CDM led to significant achievements only within 4 years.

Three lessons from abroad

UK: Results of Castlefields Health Centre pilot of active management of conditions:

- 15% reduction in admissions for older people
- average length of stay fell by 31 per cent (from 6.2 days to 4.3 days)
- total hospital bed days used by this group fell by 41 per cent
- improved links between practice staff and other agencies in the community, leading to more appropriate referrals to other services and much faster response times for social services assessments

US: Veterans Administration, focus on improving chronic disease management:

- 50% reduction in bed-day rates from 1994-1998
- 35% reduction in urgent care visit rates
- moderate increase in clinic visits, tests and Consultations

Canada: Chronic disease management program

- 7% fall in unplanned and emergency hospitalisations from 1998 to 2002)

The reasons for this marked decline is attributed to a number of interventions focussed on better management of chronic disease including; *report cards*; *knowledge management*; *guidelines and protocols for patients and physicians*; and *learning from top performing physicians*.

Fig. 1. Results of good chronic disease management in United Kingdom and United States and Canada.

As it is shown in Figure 2, according to world health organization, in Spain in 2002 around 90% of all deaths were caused by chronic disease. This emphasises on the need for good CDM in Spain.

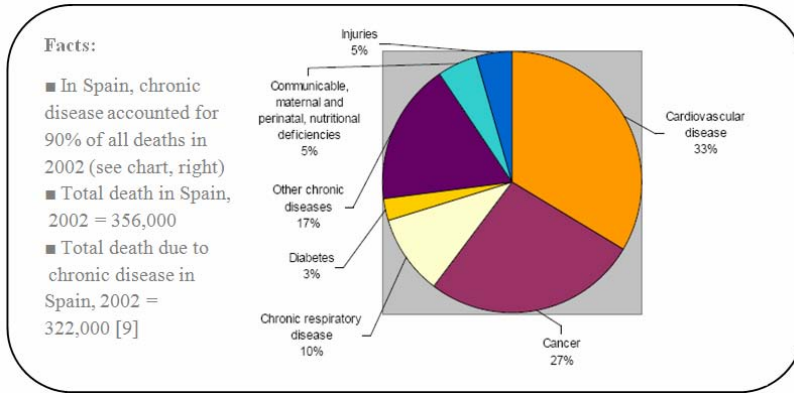


Fig. 2.
Deaths by
cause for all
ages in
Spain in
2002

3 Electronic Institutions

Multi-agent systems (MAS) are collections of autonomous intelligent entities that collaborate in the joint resolution of a complex problem. These kinds of systems are appropriate in many medical domains because of the quality of the problems in this area. Such agent-mediated medical systems can manage complex tasks and have the potential to adapt to unexpected events.

Electronic institutions [3] [8] are based on three main elements: a Dialogical Framework, a Performative Structure and a set of Normative Rules [9] [10]. However in this paper the normative rules are not discussed.

The *Dialogical Framework (DF)* defines the valid illocutions that agents can exchange as well as the language and ontology they use. Moreover, it defines the participant roles within the EI and their relationships. Each role defines a pattern of behaviour within the EI, and any participant agent is required to adopt some of them. In the context of an EI there are two types of roles, *internal* and *external* roles. The internal roles can only be played by what are called *staff* agents which are those related to the institution.

For each activity interactions between agents are articulated through agent group meetings, called *scenes*, using well-defined communication *protocols*. As a result, all agent interactions that take place in an EI exist within the context of a scene. In addition, the protocol of each scene models the possible dialogues among roles instead of agents. One of the features of the scenes is that they allow agents either to enter or to leave a scene at certain particular moments (*states*) of an ongoing conversation depending on their role. A scene protocol is specified by a directed graph, where nodes represent the different conversation states and arcs are labelled with illocution schemes or timeouts that allow the conversation state evolve. Hence, at each point of the conversation, electronic institution defines what can be said, by whom and to whom.

As a scene models a particular multi-agent dialogic activity more complex activities can be specified by establishing relationships among them illustrated in the *Performative Structure (PS)*. In other words, the activities represented by PS can be shown as a collection of multiple, concurrent scenes. In a PS agents move from scene to scene, constrained by the rules defining the relationships among the scenes. Moreover a PS

can be thought as a network of scenes in which, the *initial* and *final* scenes determine the entry and exit points of the institution respectively.

4 Hospitals Chronic Disease Management and Purchasing (HCDMP) System

HCDMP System is an electronic institution in a hospital environment using agent technology which, in order to help chronic patients provides the followings to the patient home address by post:

1. *Guidebooks*: including patient education about self management of their illness, for example helping them to understand what to do, how to adjust their medication dose, and how and when to use health care.
2. *Prompts and Reminders*: for when they should be doing something and attending for care.
3. *Support from a knowledgeable patient*: introducing knowledgeable patients (with their consent) in the same living area, and broader networks including attending practices as part of a group of patients with the same condition.

It is clear that self management is not simply a matter of providing information to patients, it is a range of different things; however according to the UK Department of Health's Economic and Operational Research division the above self care supports work the best.

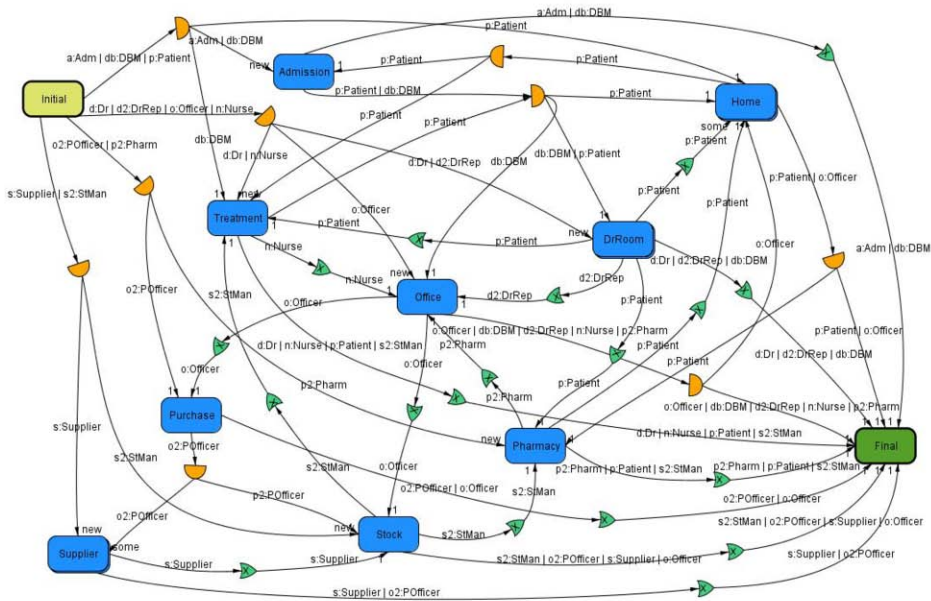


Fig. 3. HCDMP System Performative Structure: illustrates how scenes are connected through arcs and how agents move between them.

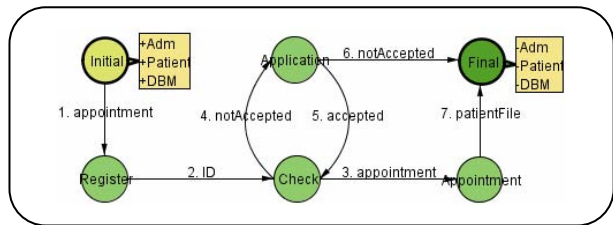
Figure 3 shows the performative structure of the HCDMP system. Scenes and roles of this electronic institution are as follow:

- Scenes: Initial, Admission, Home, DrRoom (doctor room), Office, Treatment, Pharmacy, Purchase, Stock, Supplier, Final.

- Roles: Adm (admitter), DBM (data base manager), Dr (doctor), DrRep (doctor representative), Nurse, Patient, Pharm (pharmacist), Officer, POfficer (purchase department officer), StMan (stock manager), Supplier.

Agents enrol to the Institution at the “*Initial*” scene and as it can be seen they move between the scenes depending on their roles. Initially the *Patient*, after enrolment, goes to the “*Home*” scene where all the patients, as external roles, are assumed to stay. From Home patient moves to the “*Admission*” to register and requests an appointment to visit a doctor. Figure 4 illustrates the admission scene protocol (can be thought as the inside of Admission scene) where, the *Adm* (Admitter) requests the patient for his/her *ID* (Identification number) which will be checked if it is recognised by the system as a current patient. If the ID is *accepted* and the patient is a current patient who already has a *patientFile*, then an appointment will be given to the patient to visit his/her doctor. However; if the ID is *notAccepted* then the Adm requests the patient to fill an *applicationForm* which will be processed by the Admitter. If the *applicationForm* is accepted then an appointment will be issued for the patient to visit his/her doctor, and the *DBM* (DataBaseManager) will be informed about the new *patientFile* by the Adm. Otherwise the patient is informed that he/she is notAccepted and goes to the *Final* state of the protocol to leave the Admission scene and returns to the Home scene.

Fig. 4. Shows the Admission scene where, nodes represent protocol states and arrows indicate the transitions. Also rectangles show what roles can enter or leave.



Following the performative structure, if the patient is given an appointment then goes to the “*DrRoom*” scene to visit his/her *Dr* (doctor), where the doctor in order to know the patient requests for the *patientFile* from *DBM*. Then *Dr* requests the patient *howCanIHelp* and patient informs the *Dr* from his/her problems/disease. There are two different protocols for this scene, one for the *current* hospital patients and another for the *new* patients to the hospital. Depends on the chosen protocol by the system for new or current patients, *Dr* asks different questions from the patients (i.e. about patient *diseaseHistory?* from New Patients, or if current patient *useMedicationOnTime?*, *contactOtherPatients?* and *happyWithServices?*). *Dr* then informs the patient with the *DrInstruction4P* (doctor instruction for patient) and also provides *DrInstruction4O* (doctor instruction for Office) to the *DrRep* (doctor representative). Afterwards, patient from *DrRoom* scene depends on *DrInstruction4P* moves either to Home, “*Pharmacy*” or “*Treatment*” scene. And *DrRep* from *DrRoom* moves to “*Office*” scene to inform the Officer of the *DrInstruction4O*.

In the Pharmacy scene patient requests for his/her prescription from *Pharm* (pharmacist).

However if the *Dr* diagnoses that patient needs extra care then the patient goes to the Treatment scene to stay in the hospital until the patient either fully recovers and leaves the hospital to Home, or dies.

In Office scene *DrRep* informs the *Officer* of the *DrInstruction4O* including: *PatientID*, *PatientName* and *recommendedServices* for the patient. Then officer requests the

patientFile from DBM in order to process the doctor recommendations and provide services to the patient including: *guidebooks*, *reminders*, *similarAreaPatients* (please note that the similarAreaPatient details are released by the patients consent, asked in registration form) and *recommendedOrganizations*. (recommendedOrganizations are the organizations which might be able to provide extra help for the patients. i.e. psychiatrists) Officer then moves to the Home scene to inform the patient about the services he/she will receive. Officer also moves to the “Purchase” scene to inform the POffer (purchase department officer) about the *purchaseList* (purchaseList, prepared by the Officer, is a list of requested products by the Pharmacy and Treatment department). In addition Officer has to move to the “Stock” scene to inform the StMan (stock manager) of the requested products (mainly medicine) for Pharmacy and Treatment scenes. These products will be delivered later by the StMan to the Pharmacy and Treatment scenes.

In Home scene patient is informed by the Officer about the services including similarAreaPatients details who are patients with the same disease and condition and are willing to help other patients. These patients using this information can contact each other and discuss their problems and help each other. This is highly recommended by doctors to introduce patients with similar conditions to each other. Patients then can request other patients for *Consultation* including: their *problems*, *DrInstructions* and the *services* they receive.

In Purchase scene Officer informs the POffer of the *purchaseList* including: *items*, *quantity*. POffer then informs the Officer of the *purchaseDetails* including: *price*, *productInfo*, *deliveryTime* and *quantity*.

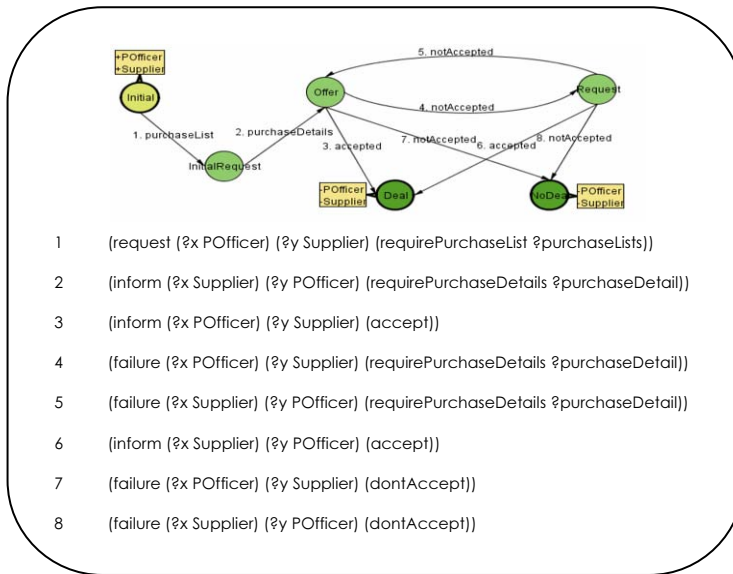


Fig. 5. Shows the Supplier scene, the protocol states and the messages using Fipa agent communication language [11].

As it can be seen in figure 5, which shows the “Supplier” scene protocol (can be thought as the inside of Supplier scene), POffer requests the purchaseList from the Supplier (message #1) and Supplier informs the POffer of the purchaseDetails (#2) including requested *items*, *price*, *productInfo*, *deliveryTime* and *quantity*. If this purchaseDetails is accepted by the POffer (#3) then the purchasing process is completed but, if the purchaseDetails is notAccepted by the POffer then it informs the Supplier of the failure to accept the purchaseDetails and requests a modified

PurchaseDetails to come to an agreement with the supplier (#4) where, if it is accepted by the Supplier (#6) then the business is done. Otherwise the Supplier informs POffer of another modified PurchaseDetails modified by the supplier (#5). If it is accepted by the POffer (#3) then the purchasing process is completed, but if it is notAccepted then the Supplier will be informed about the failure to accept the supplier offer until it sends another modified PurchaseDetails. This can be done as many times between the POffer and Supplier until they either come to an agreement (#3 or #6) or refuse to do more business (#7 or #8). In the Stock scene POffer informs the *StMan* (stock manager) of the *purchase* including *items*, *quantity* and *productInfo*. Then Supplier moves to the Stock scene in order to deliver the purchased products and informs the *StMan* of the *deliverProducts*. *StMan* then informs the POffer of the *deliverProducts*.

5 Related work

The aim of this paper was to specify a Business Process Management (BPM) setting in terms of a Multi-agent organisational environment using Islander tool. Specifically, to use the EIDE environment, developed by the Artificial Intelligence Research Institute (IIIA-CSIC), to model a business process management example. As a result Hospitals Chronic Disease Management and Purchasing System was developed using Electronic Institutions and Islander tool explained in sections 3 and 4. This approach and similar techniques are employed by *Metastorm* [12]. *Metastorm* offers market-leading software for Business Process Management (BPM).

Metastorm Manager's Edition Designer is the main tool used to create a process and the *maps* and *forms* required to define and implement the procedure. In addition Designer is used to build the business rules that are used to automate business processes. Studying *Metastorm* software illustrates that *maps* are equivalent to *performative structures* in Islander as well as *forms* equivalent to *Ontologies*. *Metastorm Designer* is used to create models of business processes in which an electronic *folder* (equivalent to *scene* in Islander) is passed from *stage* (*state*) to stage within the process. At each stage *actions* may be taken either automatically or by individuals or groups. Please note that actions are comparable to the *protocols* of Islander tool.

Also *Prometheus Design Tool* [13] similar to Islander is a graphical editor for designing agent systems. There are three design phases in *Prometheus*:

- *System specification*: in which *actors* (human or software) are identified and *roles* (identified by *goals*, *percepts* and *actions*) are described and captured.
- *High-level (architectural) design*: in which similar to the performative structure in Islander, an overall structure of the system is described using a system overview diagram. The *protocols* are defined in this phase too.
- *Detailed design*: in which the internals of each agent are developed.

Islander and *Prometheus Design Tool*, both are used to develop various aspects of specifying and designing a system using an agent oriented approach and to support the design and development of multi-agent systems.

6 Conclusions and Future work

In this paper we described how important the chronic disease management was and how a good management could make significant improvements. According to WHO about 90% of all deaths in Spain in 2002 has been due to chronic disease. This emphasises on the importance of CDM in Spain.

Agent technology is appropriate for many medical domains, including chronic disease management, as they are collections of autonomous intelligent entities that collaborate in the joint resolution of a complex problem. We introduced and explained HCDMP system. An electronic institution which can help chronic disease management and purchasing processes of the chronic departments in hospitals.

As future work we plan to implement the agents of HCDMP model and try some experiments using aBuilder (agent builder) and AMELI software.

Acknowledgements.

This work was partially funded by projects AT (CONSOLIDER CSD2007-0022), IEA (TIN2006-15662-C02-01), EU-FEDER funds, and by the Generalitat de Catalunya under grant 2005-SGR-00093. Marc Esteva enjoys a Ramon y Cajal contract from the Spanish Government.

References

1. Luck, M., McBurney, P., Shehory, O., Willmott, S. *Agent Technology: A Roadmap for Agent Based Computing*. AgentLink (ISBN 0854328459) 2005.
2. Javier Vázquez-Salceda "Normative Agents in Health Care: Uses and Challenges", *AI Communications* vol. 18 n. 3, pp. 175-189. IOS Press, 2005.
3. Marc Esteva, Juan A. Rodríguez-Aguilar, Carles Sierra, Pere Garcia and Josep L. Arcos, "On the Formal Specification of Electronic Institutions" in the book *Agent-mediated Electronic Commerce: the European AgentLink Perspective*, Lecture Notes in Artificial Intelligence, pp. 126-147, Springer-Verlag, 2001.
4. Department of Health: Improving Chronic Disease Management: http://www.dh.gov.uk/en/Publicationsandstatistics/Publications/PublicationsPolicyAndGuidance/DH_4075214.
5. World Health Organization Report: http://www.who.int/chp/chronic_disease_report/en/
6. Marc Esteva, David de la Cruz and Carles Sierra. "ISLANDER: an electronic institutions editor" in Proceedings of the First International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2002), pp. 1045-1052, Bologna, 2002.
7. Chronic Disease Management: <http://www.health.gov.bc.ca/cdm/>
8. Carles Sierra, Juan Antonio Rodríguez-Aguilar, Pablo Noriega, Marc Esteva, and Josep Lluís Arcos, "Engineering multi-agent systems as electronic institutions" in *European Journal for the Informatics Professional*, V(4) N. 4, pp. 33-39, 2004.
9. Gaertner, D., García-Camino, A., Noriega, P., Rodríguez-Aguilar, J.A., Vasconcelos, W. "Distributed norm management in regulated multi-agent systems" In Sixth International Joint Conference on Autonomous Agents and Multiagent Systems. (AAMAS'07), May 2007.
10. Jordi Campos, Maite Lopez-Sanchez, Juan A. Rodriguez-Aguilar, Marc Esteva. "Formalising Situatedness and Adaptation in Electronic Institutions". In Coordination, Organizations, Institutions and Norms in Agent Systems (COIN). Workshop at AAMAS'08 Portugal, pp.103-117, 2008.
11. Fipa agent communication language: <http://www.fipa.org/repository/aclspecs.html>
12. Metastorm BPM: <http://www.metastorm.com/products/mbpm.asp>.
13. Prometheus Tool: <http://www.cs.rmit.edu.au/agents/pdt/docs/PDT-Manual.pdf>

Categorization and Social Norms Support

Daniel VILLATORO and Jordi SABATER-MIR

IIIA, Artificial Intelligence Research Institute

CSIC, Spanish National Research Council

Bellaterra, Barcelona, Spain

{dvillatoro, jsabater}@iiia.csic.es

Abstract. Social Norms proliferate in societies as a mechanism for self-organization. This kind of norms are not enforced by a central authority and the individuals of the society are those responsible for their generation and maintenance. The maintenance process is what is known as *norm support* and is supported by several mechanisms like for example laws, social proof, dominance, etc. We believe that agent based simulation is a suitable technique for investigating this topic. In this paper we present an overview of the work we have been developing in this area. The 'Find and Share (and Exchange)' game is introduced as the environment where to prove our hypotheses on social norms. After that, we present an initial categorization on the possible social norms in our environment. Finally, some mechanisms are studied to observe its effectiveness solving the norm support problem.

Keywords. Multi-agent Systems, Social Norms, Simulation

1. Introduction and Related Work

Social norms are part of our everyday life. They help people self-organizing in many situations where having an authority representative is not feasible. On the contrary to institutional rules, the responsibility to enforce social norms is not the task of a central authority but a task of each member of the society. From the book of Bicchieri [3], the following definition of social norms is extracted: "The social norms I am talking about are not the formal, prescriptive or proscriptive rules designed, imposed, and enforced by an exogenous authority through the administration of selective incentives. I rather discuss informal norms that emerge through the decentralized interaction of agents within a collective and are not imposed or designed by an authority". Social norms are used in human societies as a mechanism to improve the behaviour of the individuals in those societies without relying on a centralized and omnipresent authority. In recent years, the use of these kinds of norms has been considered also as a mechanism to regulate virtual societies and specifically societies formed by artificial agents ([12], [15], [7]). From another point of view, the possibility of performing agent based simulation on social norms helps us to understand better how they work in human societies.

One of the main topics of research regarding the use of social norms in virtual societies is how they emerge, that is, how social norms are created at first instance. This has been studied by several authors ([2], [13], [9], [6]) who propose different factors that can influence this emergence. We divide the emergence of norms in two different stages: (a)

how norms appear in the mind of one or several individuals and (b) how these new norms are spread over the society until they become accepted social norms. We are interested in studying the second stage, the spreading and acceptance of social norms, what Axelrod [2] calls *norm support*. Our understanding of norm support deals with the problem of which norm is established as the dominant when more than one norm exists for the same situation.

Our model, in contrast to those solving coordination problems ([13], [9]), can deal with social norms that are not representable in a decision table and the rewards for following a certain norm are not known a priori. A similar approach can be found in the work of Cecconi and Parisi [4], where they also deal with a simulated resource consuming society. In their work, agents do not know beforehand how good the sets of social norms they follow are, even though the authors only consider two well differentiated sets of social norms (individual strategy or collective strategy of resource consumption). However, a society can have several (more than just two as we have already seen in the literature) sets of social norms abided by different members of the society. In the work of Sen [14], we observe that the authors present 6 different strategies (or sets of social norms), but they study the behaviour of mixed populations of these kinds of agents. Specifically, we study the situation where **while having initially different sets of social norms in a society, after some time, one of these sets (the one that maximizes the common goal of the society) prevails over the rest.**

For the sake of simplicity, we assume that all agents pursue the same global objective while trying to satisfy, as a second instance, its own objective. As we said, we want to study the situation where a single set of social norms, after some time, prevails over the rest. In order to achieve that task, we need to know beforehand the quality of a set of norms in a society, assuming that all the agents share the same set of social norms. Once a ranking of the different set of social norms is fixed, we can observe how the mechanisms we plan to apply in the norm support problem behave.

This is the first step that should allow us to study in the future more complex situations where different sets of norms sharing the same social space, with similar levels of satisfaction at the individual level, can achieve a better global result than a single dominant set. In the study presented in this paper we use a social evaluation mechanism as the *image* (which is the own believed evaluation of the others) as the main mechanism to facilitate the process of *norm support*. We also introduce the concept of ‘visionary’ individuals as a special kind of individual that by means of local simulations of the environment can partially foresee how a set of norms should work in the society if they were adopted as the dominant set.

2. Reference Scenario

In order to design an scenario where the usage of social norms is significant, we are inspired by real life examples ([11], [5]), where the usage of social norms is crucial for the survival of the society. The society we use for our experiments is a resource-gatherer distributed and decentralized society. All the members of the society survive by consuming resources that appear randomly in the environment and exchanging the resources among them by **abiding to a set of social norms**. Depending on the quality of these social norms, the society succeeds in the task of increasing the average life expectancy of its

members.

The application domain of this research is directly related to an ongoing research which is carried out by a group of archaeologists. We are presented an ancient historic society, already extinguished, known as ‘*the Yámanas*’. This society was located in Southern Argentina and is one of the groups of the societies commonly known as ‘*canoeros*’. They lived there for around 6000 years in a very hostile environment. The main success, and reason of study, of this peculiar society is their ability of auto-organization: the *Yámanas* were able to auto-organize themselves as a hunter-gatherer society. The archaeologists consider as a hypothesis that the key of success of this society was due to their strong respect for a known set of social norms (represented as a set of myths). These social norms regulated, among other behaviours, the resource exchange between the *Yámanas*. From the study of Gusinde [8], we extract that social norms for resource exchange regulation only made sense in such societies when the resources to be exchanged would appear sporadically although of a large contribution when they appear (e.g. finding a whale on the beach was a huge amount of resources but it would not happen frequently). Therefore, we adapt the parameters of the simulation to this scenario.

We want to stress that even though we inspired our simulations by the previously described society, the simulation scenario is a simplification of it. Consequently, we do not intend to affirm that the results obtained out of our simulations, as they are now, are directly applicable to real societies. Notwithstanding, the results are relevant for societies of virtual agents.

3. Statement of the Problem

The problems to be faced in the following sections are two:

- Categorizing the sets of social norms in our scenario.
- Studying the effectiveness of certain mechanisms in the norm support problem.

Firstly, the problem of the categorization is performed in order to know, as designers, how to define the experimental setting for the norm support problem and, also, to interpret the results. We perform an exhaustive analysis of every possible set of social norms in our resource-gatherer society, forcing each time all the members to share the same set of social norms. This analysis provides us with the necessary information to **establish a classification of sets of social norms depending on their quality**. The quality measure used in our experiments is the Average Life Expectancy of the agents. Having fixed the ranking, we observe the characteristics that make a set of social norms optimal, with the intention of applying this characteristics to different scenarios in Section 5.2. Secondly, and making an step forward, we relax the assumption of all agents sharing the same set of social norms. Suppose now an initial population of virtual agents where each agent possesses a set of social norms although all of them pursue the same global objective. Each agent might have a different set of norms from the rest of agents. However, from [10] we extract that “everyone conforms, everyone expects others to conform, and everyone has good reason to conform because conforming is in each person’s best interest when everyone else plans to conform”. Therefore we are interested in scenarios where agents might converge to a common and optimum set of norms, as they pursue the same objective. Different mechanisms are supposed to ease and accelerate this process when

malicious agents are present. We will focus on how *image* affects the process of norm stability.

4. Simulation Model

We use a multi-agent system for our simulation. A detailed description of this model can be found in [1]. Our experimental scenario is based on a society with no central authority where all the agents survive by consuming resources found in the environment. When two agents meet, they abide by their social norms in order to decide whether to share resources or not. The fact of donating resources provides the other agent with extra resources that make it survive for a longer period of time. When one agent exhausts its resources, it ‘dies’. After dying, agents are able to reset themselves with initial resource conditions (after recalculating its *Average Life Expectancy* (ALE)). Therefore, the goal of our agents is to improve the ALE of the society. Moreover, agents have a period of time where they can exchange their social norms in order to obtain different results. Malicious agents can lie¹ during the communication process, trying to take advantage of innocent agents. We will verify the effectiveness of some mechanisms for the convergence of the optimal set of social norms when untrusted agents coexist in the society.

The simulation is divided into three periods that will be repeated until exhausting the number of initially defined time steps:

1. **Exchange Period:** During this period agents meet and exchange their sets of social norms.
2. **Execution Period:** The interactions among agents are done always in pairs, and both agents have to choose an action when interacting in this period. This decision is taken following the *set of social norms* that each agent has internalized. The set of norms in this scenario specifies if the agent has to give or not to give resources to the other agent, depending on both agent’s resource level (*Plenty*(X) means X’s resource level ≥ 100 , *Normal*(X) means X’s resource level ≥ 26 and ≤ 99 , and *Starving*(X) means X’s resource level ≤ 25). When two agents meet, each agent is able to observe its own level of resources and its opponent’s level. Table 1 represents the whole list of possible situations (formed by two observables) in which an agent may find itself.
3. **Evaluation Period:** Agent have the option to recover the set of norms they previously have had. During this period agents will decide if they want to recover them.

5. Experiments and Results

We present two different sets of experiments in this section. Firstly, results on the categorization of social norms are presented. Secondly, some mechanisms are studied in the simulation scenario to observe their effectiveness on the norm support problem.

¹Liar agents lie by informing about the efficiency of their set of social norms saying that is 5 times better than it actually is when they communicate with other agents.

| Situation | | Action |
|--------------|---------------|-----------------|
| Starving(Me) | Starving(You) | Give / Not Give |
| Starving(Me) | Plenty(You) | Give / Not Give |
| Starving(Me) | Normal(You) | Give / Not Give |
| Plenty(Me) | Starving(You) | Give / Not Give |
| Plenty(Me) | Plenty(You) | Give / Not Give |
| Plenty(Me) | Normal(You) | Give / Not Give |
| Normal(Me) | Starving(You) | Give / Not Give |
| Normal(Me) | Plenty(You) | Give / Not Give |
| Normal(Me) | Normal(You) | Give / Not Give |

Table 1. Situations and Actions. Structure of a set of social norms.

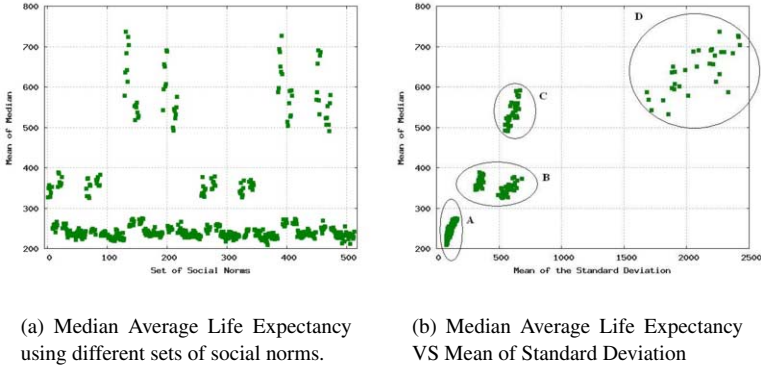


Figure 1. Categorization of Social Norms

5.1. Social Norm Categorization

In this experiment we want to fix a ranking of all the possible sets of social norms available for the agents. All the experimental parameters are fixed. The population of agents will share the same set of social norms in each simulation. As we have already said, this might seem a very ideal situation (all agents sharing the same set of social norms), but, it is the situation that the norm support problem plans to reach after solved. After running an exhaustive test over all the possible set of social norms, we can observe the results in figure 1(a). The horizontal axis represents each one of the 512 possible sets of social norms². The vertical axis represents the mean of the median average life expectancy of the society from each of the 20 simulations.

From the experimental results we can observe that under the same environmental conditions, different sets of social norms produce different results in the agents average life expectancy (ALE). In figure 1(a), we can perfectly distinguish between three different levels: the one we define as **Bad** sets of social norms (median ALE lower than 300), the one defined as **Average** sets of social norms (median ALE between 300 and 400), and the one defined as **Good** sets of social norms (median ALE higher than 400). In figure 1(a), and in the levels aforementioned, we continuously refer to the mean of the median ALE. This median ALE represents information from only one member of the society, and does not provide us with a precise idea of how the rest of the society has behaved.

²2 actions raised to the power of 9 situations gives us a result of 512 different sets of social norms that will be studied separately.

It could happen that in two different societies with the same median ALE, the distance between the best and the worst member of the society was very different: one very large, representing a heterogeneous society; and another one very small, representing a homogeneous society. In order to observe the homogeneity of each society, produced by the sets of social norms, we observe also the Average Standard Deviation of the simulations. If the Average Standard Deviation is low, this shall mean that all the agents have obtained similar results, producing consequently, an homogeneous society.

In figure 1(b), we can observe four different data clusters. These clusters represent the homogeneity of each society using a specific set of social norms. The cluster A represents the most homogeneous society, followed by B, C, and D, that is the most heterogeneous. The sets of norms that show a good (high) performance deserve a deeper study. Consequently we extract such sets of norms and analyze the characteristics of both high clusters (C and D).

The sets of norms obtained in the heterogeneous cluster share some common characteristics. These characteristics are extracted from the theory of Karnaugh Maps [16], obtaining the following generalization:

If *Plenty*(*Agent_A*) **Then** Do Not give Resources to *Agent_B*
If *Normal*(*Agent_A*) **Then** Give Resources to *Agent_B*

One conclusion that we may extract from this experiment is: when being an agent in resource-scarce environments, do not consider the others state, give only when you are normal and do not give when you are plenty of resources. This kind of norms encourages the enrichment of those that are *Plenty*, taking advantage of those that continuously die and resurrect, and not returning anything to the society. Thus, we have obtained a selfish society, but remembering that obtains good results although in an heterogeneous manner. We still have to analyze the homogeneous cluster. The norms extracted from the homogeneous-high cluster are the following:

If *Normal*(*Agent_A*) **Then** Give Resources to *Agent_B*
Else Do Not give Resources to *Agent_B*
If *Plenty*(*Agent_A*)
If *Normal*(*Agent_B*) **Then** Give Resources to *Agent_B*
Else Do Not give Resources to *Agent_B*

These norms, in contrast to the heterogeneous norms, do pay attention to the other agents state to decide the action to take. Possibly, this refinement in the decision process is the cause of the homogeneity.

5.2. Norm Support Problem

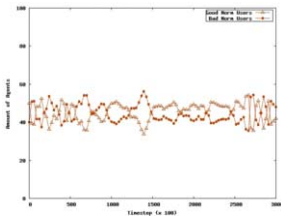
In this section we analyze the factors that make that a certain set of social norms becomes the dominant set in a society where initially, individuals were using different sets of social norms. From now on we will refer to good, average or bad agents as agents that use a good, average or bad set of norms respectively (as defined in section 5.1).

We believe that when agents are self-interested and are donated with certain mechanisms, they can self-organize themselves to the best configuration available for them. In this case, agents are able to change their social norms for other set of norms transmitted to

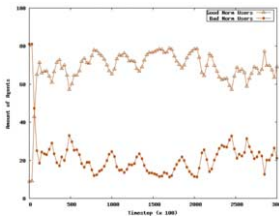
each of them. The problem arrives when we deal with an insincere population. Therefore, agents need an extra mechanism, as may be the *Image System*, to control fraudulent behaviours. *Image* is the own believed evaluation that one agent has of another agent, hence, our agents are donated with a memory of what the other agents said to them and then an internal evaluation of the degree of truth of the information transmitted to them. In case the agent source of information is a liar, our agents will block interactions with that agent by adding its identity to its own black list.

In figure 2(a) we can see how the presence of liar agents introduce a great instability into the system, making impossible for the other agents to reach any stable state. In figure 2(b) we observe how the possibility for the agents to have a certain social control allows them to stabilize better than in previous situations.

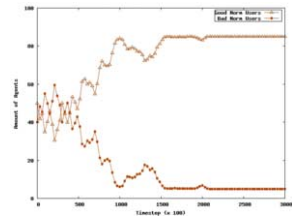
Finally, and in order to accelerate the process of convergence into the best configuration, we introduce the concept of 'Visionary' agent. This kind of agent is able to load mental simulations of the information that it receives. If any agent informs it (the visionary agent) about, for example, that the effectiveness of a set of social norms is very good, the visionary agent is able to load this mental simulation that will give it an approximate value and then compare it with the transmitted information. In this way, instead of detecting liar agents and avoid them, the visionary agent also detects the best set of social norms and ease its transmission through the rest of the society. The result can be seen in figure 2(c).



(a) Image System Off.



(b) Image System On. 9 Good VS 81 Bad.



(c) Image System On. One Good VS 99 Bad.

6. Conclusions and Future Work

We have presented in this article a simulated society and an exhaustive study of social norms aimed at sharing resources that members of such society might use. From this analysis, we have established a quality scale of the different sets of social norms when acting separately. In addition, and relaxing the hypothesis of all the agents sharing the same set of social norms, we have studied several mechanisms that ensure a society with different sets of social norms to self-organize to the best set of norms. Special attention has been paid to the *Image System* that in future versions will be extended with reputation mechanisms. Finally, the visionary agent has been presented as a mechanism that accelerates the process of norm support.

All the techniques applied to this kind of simulated self-organized society can be directly translated to real-world applications. One of these applications is the open peer-to-peer information exchange system. Social norms can help ensuring the equality of all the

members, stabilizing in the most efficient set of norms, and detecting fraudulent agents. As part of the future work, after proving that the Norm Support Process is improved by the addition of a reputation mechanism, we plan to apply the same mechanisms into a peer-to-peer information exchange system. Our long-term objective is the implementation of a fair, balanced, trusted and self-organized peer-to-peer network, through the usage of social norms, reputation theory and agent-based simulation. As another long-term objective, our research will serve as a simulation platform where to confirm some hypotheses in the archaeological field about how and what mechanisms ‘the Yámanas’ used for self-organizing .

7. Acknowledgments

This work was supported by the European Community under the FP6 programme (eRep project CIT5-028575 and OpenKnowledge project FP6-027253), by the project Autonomic Electronic Institutions (TIN2006-15662-C02-01), and partially supported by the Generalitat de Catalunya under the grant 2005-SGR-00093. Daniel Villatoro is supported by a CSIC predoctoral fellowship under JAE program.

References

- [1] Anonymous. Omitted for blinded revision process.
- [2] Robert Axelrod. An evolutionary approach to norms. *The American Political Science Review*, 80(4):1095–1111, 1986.
- [3] Cristina Bicchieri. *The Grammar of Society: The nature and Dynamics of Social Norms*. Cambridge University Press, 2006.
- [4] Federico Cecconi and Domenico Parisi. Individual versus social survival strategies. *Journal of Artificial Societies and Social Simulation*, 1(2), 1998.
- [5] Frans de Waal. Good natured. *Harvard University Press*, 1996.
- [6] Cora Beatriz Excelente-Toledo and Nicholas R. Jennings. The dynamic selection of coordination mechanisms. *Journal of Autonomous Agents and Multi-Agent Systems*, 2004.
- [7] Amandine Grizard, Laurent Vercouter, Tiberiu Stratulat, and Guillaume Muller. A peer-to-peer normative system to achieve social order. In *Workshop on COIN @ AAMAS’ 06*, 2006.
- [8] Martin Gusinde. *Los Indios de la Tierra del Fuego*. CAEA, 1982.
- [9] James E. Kittock. The impact of locality and authority on emergent conventions: initial observations. In *AAAI’94 Proceedings of the Twelfth National Conference on Artificial Intelligence*, volume 1, pages 420–425. American Association for Artificial Intelligence, 1994.
- [10] David Lewis. *Convention: A Philosophical Study*. Harvard University Press, 1969.
- [11] Mario Paolucci, Rosaria Conte, and Gennaro Di Tosto. A model of social organization and the evolution of food sharing in vampire bats. *Adaptive Behavior*, 41(3):223–239, 2006.
- [12] Nicole J. Saam and Andreas Harrer. Simulating norms, social inequality, and functional change in artificial societies. *Journal of Artificial Societies and Social Simulation*, 2(1), 1999.
- [13] Sandip Sen and Stephane Airiau. Emergence of norms through social learning. *Proceedings of IJCAI-07*, pages 1507–1512, 2007.
- [14] Sandip Sen, Anish Biswas, and Sandip Debnath. Believing others: Pros and cons, 2000.
- [15] Yoav Shoham and Moshe Tennenholtz. On the synthesis of useful social laws for artificial agent societies (preliminary report). In *Proceedings of the AAAI Conference*, pages 276–281, 1992.
- [16] Don Thompson. On constructing karnaugh maps. *SIGCSE Bull.*, 19(2):20–23, 1987.

This page intentionally left blank

Constraints, Satisfiability, and Search

This page intentionally left blank

How Hard is a Commercial Puzzle: the Eternity II Challenge¹

Carlos ANSÓTEGUI, Ramon BÉJAR, Cèsar FERNÁNDEZ, and Carles MATEU

{carlos, ramon, cesar, carlesm}@diei.udl.cat

Dept. of Computer Science, Universitat de Lleida, SPAIN

Abstract. Recently, edge matching puzzles, an NP-complete problem, have received, thanks to money-prized contests, considerable attention from wide audiences. We consider these competitions not only a challenge for SAT/CSP solving techniques but also as an opportunity to showcase the advances in the SAT/CSP community to a general audience. This paper studies the NP-complete problem of edge matching puzzles focusing on providing generation models of problem instances of variable hardness and on its resolution through the application of SAT and CSP techniques. From the generation side, we also identify the phase transition phenomena for each model. As solving methods, we employ both; SAT solvers through the translation to a SAT formula, and two ad-hoc CSP solvers we have developed, with different levels of consistency, employing several generic and specialized heuristics. Finally, we conducted an extensive experimental investigation to identify the hardest generation models and the best performing solving techniques.

1. Introduction

The purpose of this paper is to introduce a new set of problems, edge matching puzzles, a problem that has been shown to be NP-complete [8], modelling them as SAT/CSP problems. Edge matching puzzles have been known for more than a century (E.L. Thurston was granted US Patents 487797 and 487798 in 1892) and there is a number of child toys based on edge matching puzzles. These puzzles have recently received world wide attention with the publication of an edge matching puzzle with a money prize of 2 million dollars if resolved (*Eternity II*). This kind of competitions is both, a challenge to develop more competitive SAT/CSP solvers, and a real showcase to show recent advances in hard problem solving attained by the SAT/CSP community.

Our contribution is threefold. First, we provide an algorithm for generating edge matching puzzles. The proposed algorithm is simpler and faster than other generators of hard SAT/CSP instances. Second, to our best knowledge, we provide the first detailed analysis of the phase transition phenomenon for edge matching puzzles in order to locate hard/easy puzzles. Third, we provide a collection of solving methods and a wide experimental evaluation. This collection includes SAT and CSP solving techniques. The overall solving process is to encode the edge matching puzzle as a SAT instance or CSP, and then to apply a SAT or CSP solver in order to obtain a solution for the puzzle. For SAT, we provide different SAT encodings and we apply state-of-the-art preprocessors and SAT solvers. For CSP, we encode the puzzles as CSPs with both binary and higher arity constraints and solve them with state-of-the-art CSP solvers. We have also developed two

¹Research partially supported by projects TIN2006-15662-C02-02, TIN2007-68005-C04-02 and José Castillejo 2007 program funded by the *Ministerio de Educación y Ciencia*

ad-hoc solvers based on Partial Look-ahead (PLA) [7] and Maintaining Arc-Consistency (MAC) [6] algorithms, respectively. These ad-hoc solvers are enhanced with specialized heuristics and filtering algorithms to increase performance and efficiency. Another reason for using ad-hoc CSP solvers instead of standard solvers is that this way we can use an implicit encoding of the problem that is more compact than using explicit encodings as in standard solvers, as Minion [12].

2. Preliminary Definitions

Roughly described, an edge matching puzzle is a puzzle where we must place a set of tokens in a board following a simple rule. Tokens have four sides (called also half-edges), in our case for simplicity we assume square tokens, each of a different color or pattern. The rule to follow when placing tokens is that two tokens can be placed side by side iff adjacent half-edges are of the same color (or pattern), such that when placed side by side they will form an edge with a unique color. A more formal definition is as follows,

Definition 1 (Generic Edge Matching Puzzle (GEMP)) A *Generic Edge Matching Puzzle (GEMP)*, $P(n \times m, c)$ of size $n \times m$ and c colors, is a tuple (V, S) , where V is the set of variables representing cell positions on the plane, of the form, $V = \{v_{i,j}, 1 \leq i \leq n, 1 \leq j \leq m\}$. Variables in V take values from the domain S , with $S = \{(t, r) | t \in \{T\}, r \in \{R\}\}$ being R the set of possible rotations $(0^\circ, 90^\circ, 180^\circ, 270^\circ)$, T the token subset of the form $T \subset \{(x_1, x_2, x_3, x_4) | x_i \in C\}$ and C is the set of colors, $C = \{c_i, 1 \leq i \leq c\}$.

One possible variant on GEMPs is that where token rotations are not allowed, that is, all tokens must be placed **exactly** in the same orientation as they are in the puzzle specification. Actually, this last variant coincides with the Tetravex puzzle, that has been shown also to be NP-complete [22].

Definition 2 (Generic Edge Matching Puzzle Solution) A valid solution for a GEMP, $P = (V, S)$ is an assignment of values from S to all the variables in V such that for each pair of neighboring variables, the color value assigned to the adjacent half-edges between those two variables is the same.

Definition 3 (Framed GEMP (GEMP-F)) A *Framed Generic Edge Matching Puzzle (GEMP-F)*, $P(n \times m, c)$ is a Edge Matching Puzzle that includes a special color, we represent it in figure 1 as 'gray'(0), that, in all valid solutions can only appear in variables located at the frame of the puzzle, i.e., those variables in $\{v_{1,j}, v_{n,j} | 1 \leq j \leq m\} \cup \{v_{i,1}, v_{i,m} | 1 \leq i \leq n\}$ and only on the outside half-edges of those variables.

One could think on several variants of framed puzzles attending to the sets of colors employed on distinct areas of the puzzle. In this paper we deal with two types, that have a profound impact on hardness, **one-set GEMP-F** when colors can be used at any edge of the puzzle, and **two-set GEMP-F** when two disjoint sets of colors are used; one set for edges joining frame pieces and another set for any other edge. As an example take Figure 1. One can observe that colors joining frame pieces are different from the rest. As real-world puzzles (as in *Eternity II*²) are usually framed puzzles and due to the interesting effect that the frame has on hardness this work deals with GEMP-F.

During this work we study square $n \times n$ GEMP-F problems. Rectangular puzzles will probably not be harder, similarly to what happens in problems like Sudoku [2].

²In fact, *Eternity II* is a two-set GEMP-F.

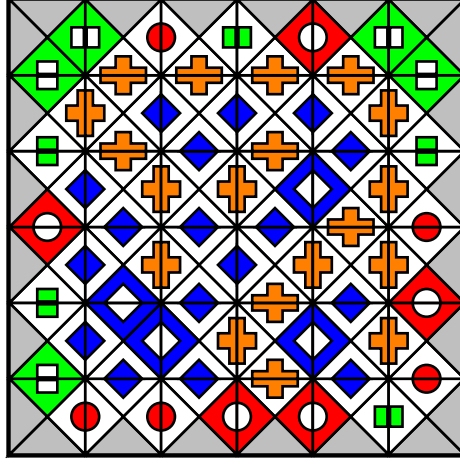


Figure 1. 6x6 size two-set GEMP-F example with 4 frame colors and 3 inner colors

Input: n, c
Output: an Edge Puzzle of size n with c colors
for $i = 1$ **to** n
 for $j = 1$ **to** n
 for $side = 1$ **to** 4
 if $v_{i,j}^{side}$ *is empty*
 $v_{i,j}^{side} = \text{random}(c)$
 $i', j' = N_{side}(i, j)$
 if $i', j' \neq 0, 0$
 $v_{i',j'}^{front(side)} = v_{i,j}^{side}$

Algorithm 1. Algorithm for generating GEMP(n, c) puzzles

3. Generation Models

The general method for a solvable puzzle generator is detailed in Algorithm 1. Roughly explained, the method assigns colors to edges of puzzle pieces (assigning a color to both half-edges). When all edges are colored, tokens are built from the existing color assignment. In the algorithm, $v_{i,j}^{side}$ refers to one of the four half-edges of the variable at position i, j , $N(v_{i,j})$ is the set of up to four neighbors of variable $v_{i,j}$ (at sides up, right, down and left), and $N_s(v_{i,j})$ gives position of neighbor at side s of $v_{i,j}$, if exists, and $0, 0$ otherwise. Finally, $front(s)$ gives the opposite side to s , that is, given two adjacent positions, s and $front(s)$ represent the two adjacent sides that join the two positions.

Special care must be taken on implementing this algorithm because this method does not prevent having repeated tokens or symmetric tokens (tokens with rotations that leave the token invariant), but for higher enough values of c (as those around the Phase Transition values), repetitions or symmetric tokens are low enough to do not suppose an impact on problem hardness.

Extending this algorithm to generate framed puzzles is easy. First the inner part of the puzzle is generated (tokens without gray color), without taking into account the frame. Then colors are assigned to the half-edges of the frame adjacent to inner tokens, that

are already determined by the inner tokens, and then half-edges that join tokens of the frame are filled with colors, randomly choosing either from the same set of colors used for the inner tokens (**one-set GEMP-F**) or from a second set of colors with no colors in common with the first set (**two-set GEMP-F**).

As it can be seen in the experimental results, this generation algorithm generates extremely hard solvable instances. The fact that the generation algorithm is so simple, in contrast with previous generation models for only-solvable structured problems like for example the one for quasigroups [1], or Sudoku [2], makes this generation model very interesting for a more detailed analysis. The most simple model for hard Satisfiable instances that we are aware of is the regular k -XORSAT [15,13], but the instances generated are not inherently hard, as even if they are hard for k -consistency based algorithms [5], they can be solved in polynomial time due to their structure based on systems of linear equations. By contrast, we do not have any guaranteed particular structure in our instances that make them easy. So, as a first step, we present in this paper an analysis of the phase transition (PT) phenomenon for the not SAT-forced version of the model, and show that the PT point coincides remarkably well with the hardest instances point of our SAT-forced model. We also show in the experimental results that similarly to what happens in [1], here the hardest instances seem to be concentrated around the point where a sudden change in the backbone variables fraction occurs.

4. Solving approaches

The following section details the methods used for solving edge matching puzzles used in this paper. We use two different approaches to the problem, solving it as a SAT formula and as a CSP. For both methods, state of the art solvers or ad-hoc solvers have been used, choosing the most efficient ones for our experimental results in the following sections.

4.1. SAT solving

The objective is to solve the edge matching puzzles through its compilation to a SAT formula and the application of a SAT solver. The immediate advantage of this approach is the availability of a wide variety of competitive SAT solvers that can be applied to our SAT encoding. However, although there has been a significant advance in the engineering of efficient SAT solvers it is still a more immature question how to design *good* encodings for a given problem.

In the following we assume, $1 \leq i \leq n$, $1 \leq j \leq n$, $t \in T$, $r \in R$, $d \in D$ and $D = \{up, right, down, left\}$.

The first SAT encoding we introduce is the *primal* encoding. The *primal* Boolean variables $p_{t,r,i,j} \in P_b$, have the following meaning: $p_{t,r,i,j}$ is true iff the token t with rotation r is *placed* at cell (i, j) . The primal constraints are the following:

- P1. A cell has exactly one token *placed* on it.

$$\bigwedge_{i,j} \left(\sum_{t,r} p_{t,r,i,j} = 1 \right)$$
- P2. A token is exactly *placed* on one cell.

$$\bigwedge_t \left(\sum_{i,j,r} p_{t,r,i,j} = 1 \right)$$
- P3. A piece matches its neighbours.

$$\bigwedge_{t,r,i,j,d} (p_{t,r,i,j} \rightarrow \bigvee_{p \in P'_b} p) \text{ such that } P'_b \text{ is the set of variables that represent the } placed \text{ tokens at the cell at direction } d \text{ from cell } (i, j) \text{ that match the color of } p_{t,r,i,j}.$$
- P4. Only the pieces at the frame can have the gray color. We write a set of unit clauses $\neg p_{t,r,i,j}$. For the pieces at the frame: t, r, i, j corresponds to a piece placed at the frame

which has not the gray color at the border. For the internal pieces: t, r, i, j corresponds to any gray colored piece placed internally.

Similarly, we could think on an encoding just working on a set of dual variables, where the dual variables represent how the edges of the puzzle are colored. The *dual* Boolean variables $e_{c,d,i,j} \in E_b$ have the following meaning: $e_{c,d,i,j}$ is true iff the edge located at cell (i, j) at direction d is *colored* with color c . Since internal edges belong to two cells, we can just use one Boolean variable to represent that an edge takes a certain color. For the sake of space, we skip the dual encoding and we present the constraints for what we call the *primal-dual* encoding:

- PD1. $P1 \wedge P2$.
- PD2. An edge is exactly *colored* with one color.
 $\bigwedge_{i,j,d} (\sum_c e_{c,d,i,j} = 1)$ Since the internal edges belong to two cells, we avoid repeating the same constraint.
- PD3. There are exactly $k_c/2$ internal edges *colored* with color c .
 $\bigwedge_c \left(\sum_{d,i,j} e_{c,d,i,j} = k_c/2 \right)$
 k_c is the number of times the color c appears at the tokens. We do not take here into account the gray color.
- PD4. If a token is *placed* on a cell then the edges have to *match*.
 $\bigwedge_{t,r,i,j} \bigwedge_{e \in E'_b} (p_{t,r,i,j} \rightarrow e)$, such that E'_b is the set of variables that represent the edges at cell (i, j) with a direction and a color that *match* the token t with rotation r at cell (i, j) .
- PD5. If an edge is *colored*, then the tokens *placed* on the cells the edge belongs to have to *match*.
 $\bigwedge_{c,d,i,j} (e_{c,d,i,j} \rightarrow \left(\bigvee_{p \in P'_b} p \right))$ such that P'_b is the set of variables that represent the tokens at cell (i, j) , with a rotation that has the color c at direction d .
- PD6. Only the edges at the frame are gray colored.
 $\bigwedge_{d,i,j} e_{gray,d,i,j}$, such that the values of d, i, j correspond to an external edge .
 $\bigwedge_{d,i,j} \neg e_{gray,d,i,j}$, such that the values of d, i, j correspond to an internal edge.

PD3 is actually a set of redundant constraints which contribute to increase the propagation power of the *complete* SAT solvers.

The above encoding channels the primal and dual encodings. Constraints PD5 and PD6 interconnect the primal and dual variables. On the one hand, they help to reduce the size of the encoding. On the other hand they increase the propagation power of SAT solvers. The level of inference we try to achieve is the one achieved by Arc Consistency in the CSP solvers, see [3].

The presented constraints have to be transformed into a conjunction of clauses. The following transformations are applied: (i) $A \rightarrow B \equiv \neg A \vee B$ where A and B are Boolean formulas and (ii) $\sum_{b \in B} b = k$ is a cardinality constraint that has to be efficiently transformed into clauses in order to keep the size of the formula as low as possible. When $k = 1$, the naive encoding has a quadratic size complexity while if we apply the transformation described in [4] we get a linear one. Similarly, when $k > 1$ we apply the default transformations applied by the pseudo-Boolean solver MiniSat+(v1.13) described in [11], which achieves a good tradeoff between the pruning power and the complexity of the encoding. Then, in order to simplify the resulting SAT formula we apply the pre-processor SatELite(v1.0) [9] with the default options. The transformation process with MiniSat+(v1.13) and the simplification with SatELite(v1.0) take less than five seconds for the hardest instances we have considered in our experimental investigation.

4.2. CSP Solving

Edge matching puzzles are easily modeled as CSP problems, with two basic sets of constraints, one set of constraints for neighboring relations, modelling the relation between half-edges and a set of global constraints modelling the fact that every token must be assigned to one variable. We have used two base algorithms for CSP solving, PLA (Partial Look-ahead) [14,7] and MAC (Maintaining Arc-Consistency), and we have added specific improvements for increasing constraint propagation. Both algorithms have been tested with two variable selection heuristics, DOM (minimum domain) and CHESSE. CHESSE is a static variable heuristic that considers all the variables of the problem as if placed in a Chess board, and proceeds by choosing all 'black' variables following a spiral shaped order from the center towards the frame, and then repeats the same procedure with 'white' variables. That causes unitary variables (singletons) appear earlier.

With the MAC algorithm we have considered the inclusion of global (n-ary) constraints with powerful filtering algorithms for maintaining generalized arc-consistency (GAC). The most important n-ary constraint we have identified is the exactly- k constraint between the set of $2k$ half-edges with a same color. That is, in any solution this set of $2k$ half edges must be arranged in a set of k disjoint pairs of half-edges. With this aim, we use the symmetric alldiff constraint (that is formally equivalent to our exactly- k constraint), and its specialized filtering algorithm [20], that achieves GAC over this constraint in polynomial time. So, we define a symmetric alldiff constraint for each exactly- k constraint we have (one for each color). More specifically, we have a color graph that represents either already matched half-edges (that become disconnected from the rest of the graph) or half-edges that could be matched given the current domains of the unassigned variables. Observe that in order to be able to extend the current partial solution to a complete solution, a necessary condition is that any color graph must contain at least one perfect matching. If this is not the case for some color, we can backtrack. Moreover, using the filtering algorithm of Regin we can eliminate any edge that will not appear in any perfect matching of the graph (i.e. to maintain GAC) and discover forced partial matchings.

We have also considered maintaining GAC for the alldiff constraint over the set of position variables using the filtering algorithm of [19].

5. Experimental Results

We present experimental results and an analytical approach for the location of the Phase Transition on one-set and two-set GEMP-F models, as well as a solver performance comparison on the instances on the peak of hardness. The hardness of GEMP-F problems is evident from the median times and from the fact that to obtain the experimental data for this section the total CPU time has been 5.5 CPU/years on a single Opteron 1.8 Ghz 64bit.

5.1. Model Hardness and Phase Transition

One-set and two-set GEMP-F present a hardness characterization depending on their constituent number of colors. As shown in Figure 2, an accurate selection of the number of colors increases the puzzle hardness by several orders of magnitude. While comparing results for one-set and two-set GEMP-F, it is worth to note that for the same puzzle size, two-set GEMP-F are harder.

As detailed in [1], one can link this hardness characterization, on only satisfiable problems, with a phase transition effect when the backbone is considered, i.e. the number of variables that take the same value on all the solutions [17]. Figure 3 shows this phase transition plotting the fraction of the backbone as a function of the number of inner colors (c_m) for two-set GEMP-F with 3 frame colors ($c_f = 3$).

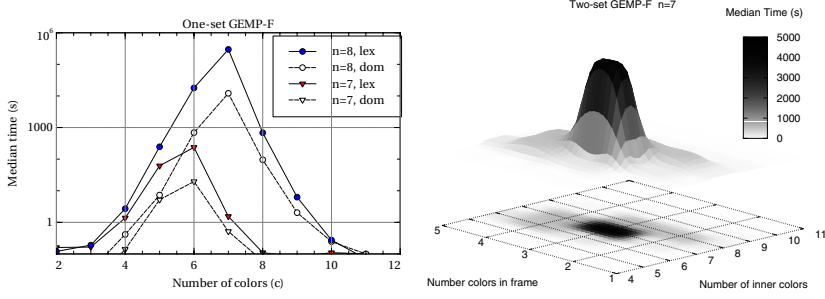


Figure 2. Hardness characteristic for a one-set GEMP-F as a function of the number of colors (left). Hardness characteristic for a 7x7 two-set GEMP-F as a function of the number of colors. Minimum median time of PLA-CHESS and PLA-DOM (right)

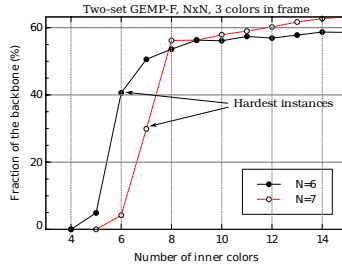


Figure 3. Phase transition of the percentage of backbone variables for a two-set GEMP-F

From an analytical point of view, we can derive some expressions that predict the phase transition location. For the sake of tractability, we consider tokens generated randomly, disregarding adjacency constraints that give only SAT puzzles. Of course, this is only an approach, but experimental results and numerical evaluations agree for both models. As usual in SAT/UNSAT models, the point where the expected number of solutions ($E[X]$) is small, but not negligible, marks the phase transition [21,18] for random CSP problems, being proved by [23] that such a transition occurs for $E[X] = 1$ on Model RB. Of course, we have not the same level of granularity on GEMP problems than in Random CSP models, and we are not able to tune our parameters to lead $E[X]$ to a desired point, but we can observe in Table 1 how the point where $E[X]$ changes from many to few solutions predicts where the harder instances are. Appendix 1 shows in detail the computation for the first moment of the number of solutions for one-set and two-set GEMP-F. It is worth to note that for $n = 16$ and $c_f = 5$ the predicted phase transition occurs at $c_m = 17$ that is exactly the number of inner colors of the two-set GEMP-F puzzle used in Eternity II contest.

Table 1 shows that hard instances may be found for one or two contiguous values of c_m , meaning that their respective median times to solve are equivalent. That is usual for small orders, tending to disappear for larger n and therefore concentrating their hard problems for a given value of c_m . Actually, using Markov inequality that gives an upper bound to the probability of having a satisfiable instance, $P(\text{Sat}) \leq E[X]$, it can be shown that $\lim_{n \rightarrow \infty} P(\text{Sat}) = 0$ beyond a critical value of $c_m > c_{m_{cr}}$. From Equations 1 we obtain that $c_{m_{cr}} = \frac{2n}{\sqrt{e}}$. Similar result apply for one-set GEMP-F.

Table 1. Round of $\log_{10}(E[X])$ according to Eq. 1 for two-set GEMP-F. Shadowed cells shows where the hardest problems have been experimentally found

| $\mathbf{c_f} \setminus \mathbf{c_m}$ | $\mathbf{n = 6}$ | | | | $\mathbf{n = 7}$ | | | | | $\mathbf{n = 8}$ | | |
|---------------------------------------|------------------|----|----|----|------------------|----|----|----|-----|------------------|----|----|
| | 4 | 5 | 6 | 7 | 4 | 5 | 6 | 7 | 8 | 6 | 7 | 8 |
| 3 | 4 | 0 | -3 | -6 | 12 | 7 | 2 | -2 | -6 | 10 | 4 | -1 |
| 4 | 2 | -2 | -6 | -8 | 9 | 4 | -1 | -5 | -9 | 6 | 1 | -4 |
| 5 | | | | | 7 | 1 | -3 | -7 | -11 | 3 | -2 | -7 |
| 6 | | | | | 5 | -1 | -5 | -9 | -13 | 1 | -4 | -9 |

Table 2. Comparison of solving approaches for the one-set and two-set models. 100 instances per point.

| Size ($n \times n$) | One-set GEMP-F | | Two-set GEMP-F | | | | | |
|--------------------------|------------------------------|-----------------|----------------|-----------------|-----------------|-----------------|------------|-----------------|
| | 7 × 7 | 8 × 8 | 6 × 6 | 7 × 7 | | | | |
| Colors inner[:frame] | 6 | 7 | 6:2 | 6:4 | 7:2 | 7:3 | 7:4 | 8:2 |
| PLA-LEX | 235 | $>2 \cdot 10^5$ | - | - | - | - | - | - |
| PLA-DOM | 20 | 12,125 | 15 | 520 | 18,193 | 9,464 | 581 | 8,387 |
| PLA-CHESS | 42 | 52,814 | 0.5 | 5,249 | 137 | 4,181 | 6,906 | 510 |
| MAC+GAColor | 90 | 23,210 | 0.94 | 328 | 96 | 646 | 348 | 208 |
| MAC+GAColor+CTadiff | 92 | 22,442 | 0.73 | 377 | 94 | 727 | 395 | 216 |
| SAT(P) | 1,341 | $>2 \cdot 10^5$ | 7.45 | $>2 \cdot 10^4$ | 4,418 | $>2 \cdot 10^4$ | 7,960 | 6,465 |
| SAT(PD) | 34 | 117,823 | 0.55 | 777 | 125 | 1,785 | 682 | 359 |
| MAC _b dom/deg | 154 | 39,742 | 19 | 2,415 | $>2 \cdot 10^4$ | $>2 \cdot 10^4$ | 3,307 | $>2 \cdot 10^4$ |
| Minion | 413 | $>2 \cdot 10^5$ | 125 | 3,463 | $>2 \cdot 10^4$ | $>2 \cdot 10^4$ | 4,675 | $>2 \cdot 10^4$ |
| Solver | Median Time (seconds) | | | | | | | |

5.2. SAT and CSP Solving Methods

We generated the SAT instances according to the two previously described SAT encodings. The complete SAT solvers we experimented with were: Minisat2 (v.061208-simp) [10], siege(v.4.0), picosat(v.535) and satz [16]. Minisat2 was the best performing SAT solver, and required to activate the option *polarity-mode=true*. For the state-of-the-art CSP solvers, Minion and MAC_b dom/deg[6], we adapted the primal and primal-dual encodings taking into account variables with a domain greater than or equal to two (we only report results for the best encoding).

Table 2 shows median time results for one-set and two-set GEMP-F with distinct sizes and number of colors, solved with several techniques. These techniques are: (i) PLA CSP solvers with variable selection heuristics LEX, DOM and CHESS, explained above; (ii) MAC with filtering algorithm for Generalized Arc-Consistency for color graphs (GAColor), using CHESS heuristic, with and without GAC filtering for the alldiff over position variables (CTadiff); (iii) the Minisat2 (v.061208-simp) [10] on the primal encoding SAT(P), and on the primal-dual encoding SAT(PD), and (iv) the state-of-the-art CSP solvers Minion and MAC_b dom/deg.

On one hand, the best performer for one-set GEMP-F is PLA-DOM meanwhile for two-set puzzles MAC+GAColor is the best one. It seems that the additional pruning effect of the GAColor filtering is powerful enough to pay off the additional time needed by such filtering in the two-set GEMP-F.

On the other hand, on PLA solvers for two-set GEMP-F, CHESS heuristic performs better than DOM when the number of frame colors is lower, and this could be because CHESS instantiates frame variables at the end of the search, and in those cases, the probability of finding a consistent frame is higher than when the number of frame colors is

higher. About SAT solvers, the best performing encoding is the primal-dual encoding being quite competitive with the CSP approaches, but still with a poor scaling behaviour.

6. Conclusions

This work clearly shows that edge matching puzzles are a very hard problem with a reduced and simple definition and a very easy and fast generation process. State of the art solvers (SAT or CSP) cannot solve problems bigger than a meager 8×8 . Even using sophisticated specialised filtering algorithms, solvers are unable to keep pace with the problem hardness scaling. This makes GEMP-F a really challenging problem.

Appendix 1

In this appendix, we derive exact expressions to the number of solutions of one-set and two-set GEMP-F, when tokens are generated at random, unregarding adjacency constraints that give only SAT puzzles.

For a two-set GEMP-F, according to Definition 1, one can think on set T as $T = T_c \cup T_f \cup T_m$, being T_c , T_f and T_m the set of tokens corresponding to the corners, rest of the frame and mid of the board respectively.

Lets denote as $\mathcal{S} = \mathcal{S}_c \times \mathcal{S}_f \times \mathcal{S}_m$ the set of possible locations on the board for T_c , T_f and T_m jointly, and \mathcal{C} the subset of \mathcal{S} that satisfies 2-set GEMP-F rules. Clearly, considering a $n \times n$ board, and that only elements of the set T_m can be rotated:

$$|T_c| = 4, \quad |T_f| = 4(n-2), \quad |T_m| = (n-2)^2 \\ |S_c| = 4!, \quad |S_f| = (4(n-2))!, \quad |S_m| = 4^{(n-2)^2} \cdot ((n-2)^2)!$$

We define X as the random variable that denotes the number of satisfying locations according to the rules of 2-set GEMP-F puzzles, (i.e. the elements of \mathcal{C}). So, its expectation can be expressed as

$$E[X] = E \left[\sum_{\sigma \in \mathcal{S}} \mathbf{1}_{\mathcal{C}}(\sigma) \right] = \sum_{\sigma \in \mathcal{S}} E[\mathbf{1}_{\mathcal{C}}(\sigma)] \\ = \sum_{\sigma_c \in \mathcal{S}_c} \sum_{\sigma_f \in \mathcal{S}_f} \sum_{\sigma_m \in \mathcal{S}_m} E[\mathbf{1}_{\mathcal{C}}(\sigma_c \times \sigma_f \times \sigma_m)] \\ = 4! \cdot (4(n-2))! \cdot 4^{(n-2)^2} \cdot (n-2)^2! \cdot E[\mathbf{1}_{\mathcal{C}}(\sigma_c \times \sigma_f \times \sigma_m)],$$

being $\mathbf{1}_A(x)$ the indicator function, i.e., takes value 1 if $x \in A$ and 0 if $x \notin A$. We claim that $E[\mathbf{1}_{\mathcal{C}}(\sigma_c \times \sigma_f \times \sigma_m)]$ is the probability that a given arrangement of tokens satisfies a 2-set GEMP-F puzzle. If tokens are build randomly, such a probability is

$$E[\mathbf{1}_{\mathcal{C}}(\sigma_c \times \sigma_f \times \sigma_m)] = \left(\frac{1}{c_f} \right)^{4(n-1)} \cdot \left(\frac{1}{c_m} \right)^{2(n-1)(n-2)},$$

being c_f and c_m the number of colors in frame and mid, respectively, and giving

$$E[X] = 4! \cdot (4(n-2))! \cdot 4^{(n-2)^2} \cdot (n-2)^2! \cdot \left(\frac{1}{c_f} \right)^{4(n-1)} \cdot \left(\frac{1}{c_m} \right)^{2(n-1)(n-2)} \quad (1)$$

Analogously, one can derive an exact expression for one-set GEMP-F, resulting

$$E[X] = 4! \cdot (4(n-2))! \cdot 4^{(n-2)^2} \cdot (n-2)! \cdot \left(\frac{1}{c}\right)^{2n(n-1)}, \quad (2)$$

where c is the number of colors.

References

- [1] Dimitris Achlioptas, Carla Gomes, Henry Kautz, and Bart Selman. Generating satisfiable problem instances. In *Proceedings of the AAAI 2000*, pages 256–261. AAAI Press / The MIT Press, 2000.
- [2] C. Ansótegui, R. Béjar, C. Fernández, C. Gomes, and C. Mateu. The impact of balance in a highly structured problem domain. In *Proceedings of the AAAI 2006*, pages 438–443. AAAI Press / The MIT Press, 2006.
- [3] Carlos Ansótegui, Alvaro del Val, Iván Dotú, Cèsar Fernández, and Felip Manyà. Modelling choices in quasigroup completion: SAT vs CSP. In *Proceedings of the AAAI 2004*. AAAI Press / The MIT Press, 2004.
- [4] Carlos Ansótegui and Felip Manyà. Mapping many-valued CNF formulas to boolean CNF formulas. In *Proceedings of the International Symposia on Multiple-Valued Logic*, pages 290–295, 2005.
- [5] Albert Atserias, Andrei A. Bulatov, and Víctor Dalmau. On the power of k -consistency. In *Automata, Languages and Programming, 34th International Colloquium, ICALP 2007*, volume 4596 of *Lecture Notes in Computer Science*, pages 279–290. Springer, 2007.
- [6] Christian Bessière and Jean-Charles Régin. MAC and combined heuristics: Two reasons to forsake FC (and CBJ?) on hard problems. In *Principles and Practice of Constraint Programming*, pages 61–75, 1996.
- [7] Rina Dechter. *Constraint Processing*. Morgan Kaufmann, 2003.
- [8] Erik D. Demaine and Martin L. Demaine. Jigsaw puzzles, edge matching, and polyomino packing: Connections and complexity. *Graphs and Combinatorics*, 23(s1):195, 2007.
- [9] Niklas Eén and Armin Biere. Effective preprocessing in SAT through variable and clause elimination. In *Proceedings of SAT 2005*, pages 61–75, 2005.
- [10] Niklas Eén and Niklas Sörensson. An extensible SAT-solver. In *Proceedings of SAT 2003*, volume 2919 of *Lecture Notes in Computer Science*, pages 502–518. Springer, 2003.
- [11] Niklas Eén and Niklas Sörensson. Translating pseudo-boolean constraints into SAT. *Journal of Satisfiability*, 2:1–26, 2006.
- [12] Ian P. Gent, Christopher Jefferson, and Ian Miguel. Watched literals for constraint propagation in minion. In *Principles and Practice of Constraint Programming*, pages 182–197, 2006.
- [13] Harri Haanpää, Matti Järvisalo, Petteri Kaski, and Ilkka Niemelä. Hard satisfiable clause sets for benchmarking equivalence reasoning techniques. *Journal on Satisfiability, Boolean Modeling and Computation*, 2(1-4):27–46, 2006.
- [14] R. M. Haralick and G. L. Elliott. Increasing tree search efficiency for constraint satisfaction problems. *AI Journal*, 14:263–313, 1980.
- [15] Matti Järvisalo. Further investigations into regular XORSAT. In *Proceedings of the AAAI 2006*. AAAI Press / The MIT Press, 2006.
- [16] Chu Min Li and Anbulagan. Heuristics based on unit propagation for satisfiability problems. In *Proceedings of the International Joint Conference on Artificial Intelligence, IJCAI 97*, pages 366–371. Morgan Kaufmann, 1997.
- [17] Rémi Monasson, Riccardo Zecchinna, Scott Kirkpatrick, Bart Selman, and Lidror Troyansky. Determining computational complexity from characteristic phase transitions. *Nature*, 400:133–137, 1999.
- [18] P. Prosser. An empirical study of phase transitions in binary constraint satisfaction problems. *AI Journal*, 81:81–109, 1996.
- [19] Jean-Charles Régin. A filtering algorithm for constraints of difference in CSPs. In *Proceedings of the AAAI 1994*, pages 362–367. AAAI Press / The MIT Press, 1994.
- [20] Jean-Charles Régin. The symmetric alldiff constraint. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, IJCAI 99*, pages 420–425. Morgan Kaufmann, 1999.
- [21] B. Smith and M. Dyer. Locating the phase transition in binary constraint satisfaction problems. *Artificial Intelligence*, 81:155–181, 1996.
- [22] Yasuhiko Takenaga and Toby Walsh. Tetravex is NP-complete. *Information Processing Letters*, 99(5):171–174, 2006.
- [23] K. Xu and W. Li. Exact phase transition in random constraint satisfaction problems. *Journal of Artificial Intelligence Research*, 12:93–103, 2000.

Random SAT Instances à la Carte ¹

Carlos ANSÓTEGUI^a, Maria Luisa BONET^b and Jordi LEVY^b

^a DIEI, UdL, Lleida, Spain. carlos@diei.udl.cat

^b LSI, UPC, Barcelona, Spain. bonet@lsi.upc.edu

^c IIIA, CSIC, Bellaterra, Spain. levy@iiia.csic.es

Abstract. Many studies focus on the generation of hard SAT instances. The hardness is usually measured by the time it takes SAT solvers to solve the instances. In this preliminary study, we focus on the generation of instances that have computational properties that are more similar to real-world instances. In particular, instances with the same degree of difficulty, measured in terms of the tree-like resolution space complexity. It is known that industrial instances, even with a great number of variables, can be solved by a clever solver in a reasonable amount of time. One of the reasons may be their relatively small space complexity, compared with randomly generated instances.

We provide two generation methods of k-SAT instances, called geometrical and the geo-regular, as generalizations of the uniform and regular k-CNF generators. Both are based on the use of a geometric probability distribution to select variables. We study the phase transition phenomena and the hardness of the generated instances as a function of the number of variables and the base of the geometric distribution. We prove that, with these two parameters we can adjust the difficulty of the problems in the phase transition point. We conjecture that this will allow us to generate random instances more similar to industrial instances, of interest for testing purposes.

Keywords. Random SAT Models, Satisfiability

Introduction

SAT is a central problem in computer science. Many other problems in a wide range of areas can be solve by encoding them into boolean formulas, and then using state-of-the-art SAT solvers. The general problem is NP-complete in the worst case, and in fact a big percentage of formulas (randomly generated instances) need exponential size resolution proofs to be shown unsatisfiable [CS88,BSW01,BKPS02]. Therefore, solvers based on resolution need exponential time to decide their unsatisfiability. Nevertheless, state-of-the-art solver have been shown of practical use working with *real-world* instances. As a consequence the development of these tools has generated a lot of interest.

The celebration of SAT competitions has become an essential method to validate techniques and lead the development of new solvers. In these competitions there are three categories of benchmarks, randomly generated, crafted, and industrial instances. It

¹Research partially supported by the research projects TIN2007-68005-C04-01/02/03 and TIN2006-15662-C02-02 funded by the CICYT. The first author was partially supported by the José Castillejo 2007 program funded by the *Ministerio de Educación y Ciencia*.

is difficult for solvers to perform well on all of them. This has lead researchers to say that randomly generated and industrial instances are of distinct nature. It has been postulated that real-world or industrial instances have a *hidden structure*. In [WGS03] it is proved that real-world formulas contain a small number of variables that, when instantiated, make the formula easy to solve. In [ABLM08] it is shown that real-world instances have a smaller tree-like resolution space complexity than randomly generated instances with the same number of variables.

The practical applicability of SAT solvers forces them to try to be good in the industrial category. However the number of benchmarks in this category is limited. Also, the instances are isolated ones, not a family of instances, one for every number of variables. And finally, they do not have a parameterized degree of difficulty. On the other hand, random formulas can be easily generated with any size, hence with the desired degree of difficulty. Moreover, they can be generated automatically on demand, what makes their use in competitions more fair, because they are not known in advance by participants. It would be interesting to be able to generate instances with the good properties of both categories.

This project was stated in “Ten Challenges in Propositional Reasoning and Search” [SKM97] and in “Ten Challenges Redux : Recent Progress in Propositional Reasoning and Search” [KS03] as the tenth challenge:

Develop a generator for problem instances that have computational properties that are more similar to real-world instances[...] Many SAT benchmarks today are encodings of bounded-model checking verification problems. While hundreds of specific problems are available, it would be useful to be able to randomly generate similar problems by the thousands for testing purposes.

Also Rina Dechter in [Dec03] proposes the same objective.

In this paper we want to make a contribution in this direction. Since we want to generate as many instances as needed, we define generators of random formulas. There are two models of generators of random formulas, the uniform and the regular random k -CNF generators. The first one has been studied for a long time and consists in selecting uniformly and independently m clauses from the set of all clauses of size k on a given set of n variables. The second one is studied in [BDIS05] and consists in selecting uniformly one formula from the set of formulas with m clauses of size k , and n variables, where all literals have nearly the same number of occurrences, i.e. either $\lfloor \frac{k \cdot m}{2n} \rfloor$ or $\lfloor \frac{k \cdot m}{2n} \rfloor + 1$. We generalize these two models by selecting variables following a geometric distribution, in the first case, and by allowing a power decreasing number of occurrences of the variables in the second case.

Another interesting feature of the formulas we generate with either of our models, is that they are around the phase transition point. This means that, for the set of instances with the ratio clauses/variables around the phase transition point, the number of satisfiable formulas that we generate is approximately the same as the number of unsatisfiable ones. We are interested in generating formulas with this precise ratio, because otherwise we could easily produce trivially satisfiable or unsatisfiable instances. Instead we want to obtain reasonably easy formulas, as close to industrial ones as possible, but not trivial as for instances that would have in them a small unsatisfiable core, for instance.

Finally, we would like to mention another property that our instances have. They can be easily parameterized in terms of their hardness. As a notion of hardness we use the space of tree-like resolution, a very close notion to the logarithm of the size of tree-like resolution. The idea is that we can fix in advance the hardness, and then we can obtain as many formulas as we want of that hardness by playing with the number of variables and other parameters of our models of random formula generation.

1. Description of the Models

1.1. Geometric k -CNF

The classical uniform k -CNF random model uses a uniform probability distribution to select the variables used in the clauses. In this subsection we propose a generalization of this model by using other probability distributions. The geometric k -CNF model is a generalization of the uniform k -CNF model where we use an exponential probability distribution with base b . Geometric k -CNF formulas may be generated with the algorithm 1.

Input: n, m, k, b
Output: a k -SAT instance with n variables and m clauses
 $F = \emptyset$;
for $i = 1$ **to** m **do**
 repeat
 $C_i = \square$;
 for $j = 1$ **to** k **do**
 $c = \text{rand}()$;
 $v = 0$;
 while $c > \text{Pr}(v; b, n)$ **do**
 $v = v + 1$;
 $c = c - \text{Pr}(v; b, n)$;
 endwhile
 $C_i = C_i \vee (-1)^{\text{rand}(2)} \cdot v$;
 endfor
 until C_i is not a tautology or simplifiable
 $F = F \cup \{C_i\}$
endfor

Algorithm 1. Geometric k -CNF generator. Function $\text{rand}()$ return a real random number uniformly distributed in $[0,1)$, and $\text{rand}(2)$ returns either 0 or 1 with probability 0.5.

1.2. Geo-regular k -CNF

In regular k -CNF formulas all literals occur nearly the same number of times, i.e. $\lfloor \frac{k \cdot m}{2n} \rfloor$ or $\lfloor \frac{k \cdot m}{2n} \rfloor + 1$ times. In geo-regular k -CNF we want them to occur a deterministic number of times, but with a frequency given by $P(X = i; n, b)$. Geo-regular k -CNF formulas are generated as follows. We construct a multiset *bag* with approximately

$Pr(X = v; B, n)^{\frac{km}{2}}$ copies of the literals v and $\neg v$, for each variable $v = 0, \dots, n - 1$. Then, we make a random partition of bag into m subsets (clauses) of size k , such that none of these clauses is a tautology or is simplifiable. Algorithm 2 describes this procedure. Notice that, when a tautology or simplifiable clause is generated, we discard all generated clauses, not just the last one.

Input: n, m, k, b
Output: a k -SAT instance with n variables and m clauses
 $bag = \emptyset$;
for $v = 1$ **to** n **do**
 $bag = bag \cup \{\lfloor Pr(v; B, n)^{\frac{km}{2}} \rfloor \text{ copies of } v\}$;
 $bag = bag \cup \{\lfloor Pr(v; B, n)^{\frac{km}{2}} \rfloor \text{ copies of } \neg v\}$;
endfor
 $S = \text{subset of } km - |bag| \text{ literals from } \{1, \dots, n, \neg 1, \dots, \neg n\}$
 maximizing $Pr(v; b, n)^{\frac{km}{2}} - \lfloor Pr(v; b, n)^{\frac{km}{2}} \rfloor$
 $bag = bag \cup S$;
repeat
 $F = \emptyset$;
 for $i = 1$ **to** m **do**
 $C_i = \text{random sub-multiset of } k \text{ literals from } bag$
 $bag = bag \setminus C_i$
 $F = F \cup \{C_i\}$;
 endfor
until F does not contain tautologies or simplifiable clauses
Algorithm 2. Geo-regular k -CNF generator

2. Experimental Results

In this section we present a series of experimental results on the location of the phase transition point and the hardness of unsatisfiable instances.

The phase transition point divides the space of CNF formulas into two regions: the under-constrained region below the phase transition point, and the over-constrained region above it. We recall two interesting properties about the phase transition point. First, around half of the random generated instances at the phase transition point are unsatisfiable, this percentage decreases across the under-constrained region and increases across the over-constrained region. The variation in the percentage of unsatisfiable instances around the phase transition point gets sharper as the number of variables increases. So, for a big number of variables, if known, the phase transition point can be used as a precise predictor of satisfiability. Second, the hardest problems empirically seem to be found near the phase transition point. Therefore, it makes sense to test candidate algorithms on these problems. However, there is not yet a method for predicting the phase transition point. It is known that for 3-SAT, for the uniform generation method, the phase transition point is located around the clause/variable ratio 4.25 and, for the regular model the ratio is around 3.55.

We measure the hardness of unsatisfiable instance as the tree-like resolution space complexity, as proposed in [ABLM08]. Contrarily to the cpu time, or number of nodes,

etc. this measure is independent of the solver/machinery used in the experiments. This measure is the minimum Strahler of a tree-like resolution tree. The Strahler is defined for binary trees as follows

$$hs(\bullet) = 0$$

$$hs(f(T_1, T_2)) = \begin{cases} hs(T_1) + 1 & \text{if } hs(T_1) = hs(T_2) \\ \max\{hs(T_1), hs(T_2)\} & \text{otherwise} \end{cases}$$

where \bullet is a node and T_1 and T_2 are trees.

The space complexity is hard to compute, therefore we approximate it as the upper-bound given by the Strahler of the search tree of an execution of an extended version of the SAT solver *satz* [LA97]. Due to the kind of lookahead performed by *satz*, we consider the leaves to have an Strahler number of 2. In [ABLM08], it is said that, for small formulas, this upper bound coincides with the real space complexity.

Each data point presented at the following figures represents the results on the computation of 200 instances, generated either by the geometric or geo-regular methods. The length of the clauses, k , was set to 3. As we have said, for solving the instances we have used an extended version of the SAT solver *satz* [LA97]. We have run the experiments on a 1Ghz machine with 1Gbyte of RAM. All the instances took at most 3000 seconds to be solved.

2.1. The Location of the Phase Transition Point

We have also identified the existence of a phase transition point for the two proposed generation methods, geometric and geo-regular, for different values of the base $b \in \{1, 2, 4, 6, 16\}$. Figure 1, shows the clause/variable ratio at the phase transition points. Notice that for $b = 1$, the geometric and geo-regular are the uniform and regular random k -CNF generators, respectively.

For the geometric (geo-regular) method we computed the phase transition points varying the variables from 40 to 300 (140 to 420), incrementing by 20. According to the number of variables the mean execution time varies from less than one second to 3000 seconds. The phase transition point, for each number of variables, was determined by selecting the clause/variable ratio that produced a number of unsatisfiable clauses closest to 50%. That numbers varied from 48.5% to 51.5%. We think that for a larger number of instances we would get number closer to 50%. Finally, in order to provide the ratio at the phase transition point, for each value of b , we computed the mean for each number of variables. As we can see in Figure 1, for both models as we increase b the clause/variable ratio decreases.

The second experiment we have conducted, shows the percentage of unsatisfiable instances as a function of the clause/variable ratio, for different values of b , and for the two models. In one case the number of variables was fixed to 420, and we varied the number of clauses by 4. In the other case we did the same with the number of variables fixed to 300. It is known that the satisfiability/unsatisfiability transition becomes sharper as we increase the number of variables. However, as we can see in Figure 2, the sharpness does not seem to depend on the value of b .

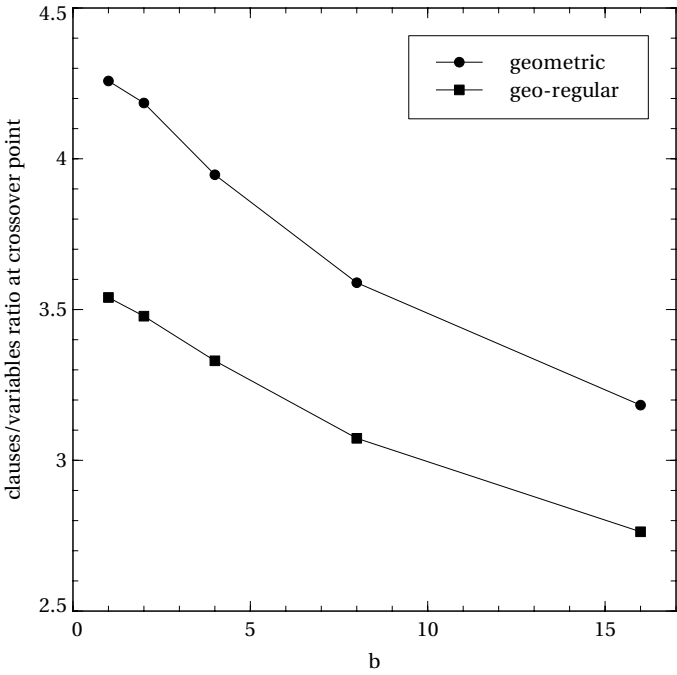


Figure 1. Clause/variable ratio at the phase transition point as a function of b .

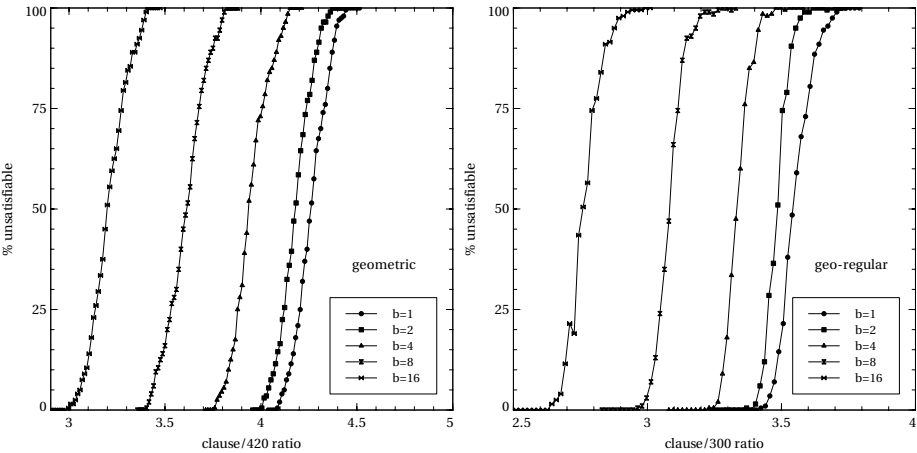


Figure 2. Percentage of unsatisfiable formulas as a function of the clause/variable ratio, with fixed number of variables.

2.2. Problem Hardness at the Phase Transition Point

As reported in previous studies, the hardest instances are usually located at the phase transition region. We decided to study the hardness of the unsatisfiable instances as a

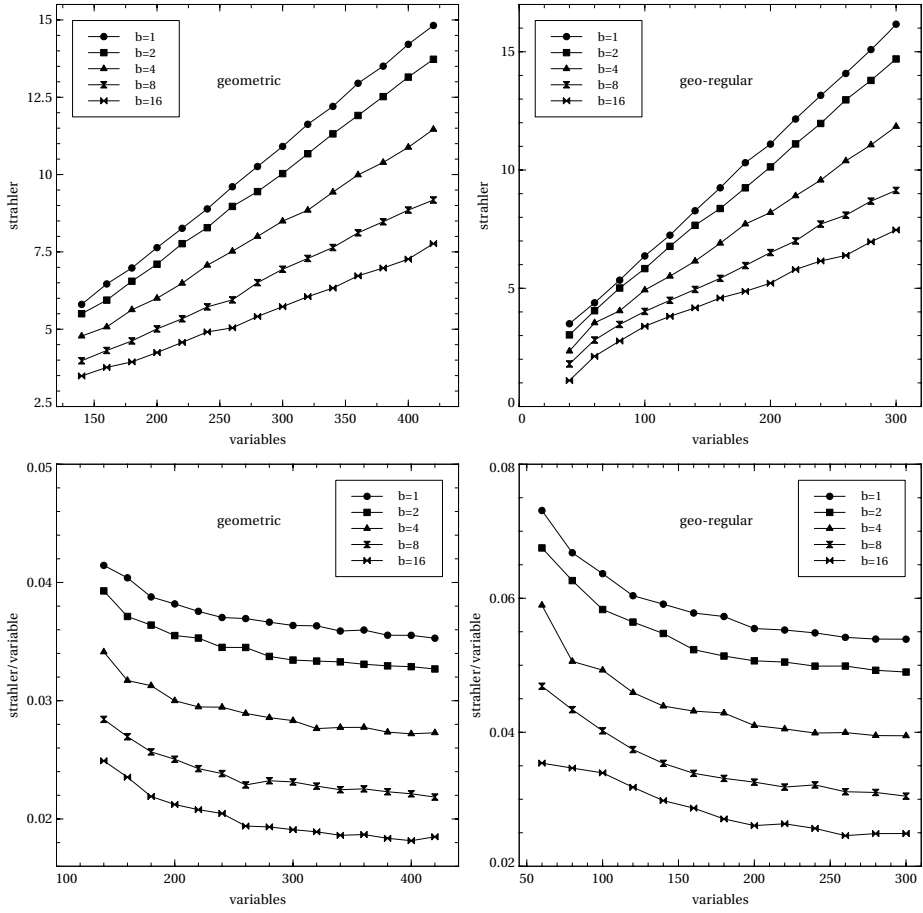


Figure 3. Strahler and strahler/variable ratio as a function of the number of variables.

function of b and the number of variables for the geometric and geo-regular methods.

There are two reasons for studying the hardness at this point. First, we plan to use our problem generators in this region, because it is where we expect to get random problems more similar to the industrial instances. These instances are (almost) minimally unsatisfiable or have very few models, and the problems generated in this region are expected to have also these properties. Second, notice that the hardness is only computed for unsatisfiable instances. As discussed in [ABLM08], there are several possible definitions for the hardness of satisfiable formulas. However, in the phase transition point, the hardness mean computed over satisfiable formulas (for a reasonable definition of hardness of satisfiable formula) and the mean computed over unsatisfiable formulas coincide.

Figure 3 shows the results. Each data point was generated as described in the previous subsection. At the first row of figure 3, we can see that for both models the hardest setting is for $b = 1$, as expected, and the strahler number grows linearly with the number of variables. For larger b 's, the growth becomes smoother. Same behavior is observed for the geometric and geo-regular methods, although geo-regular clearly dominates in

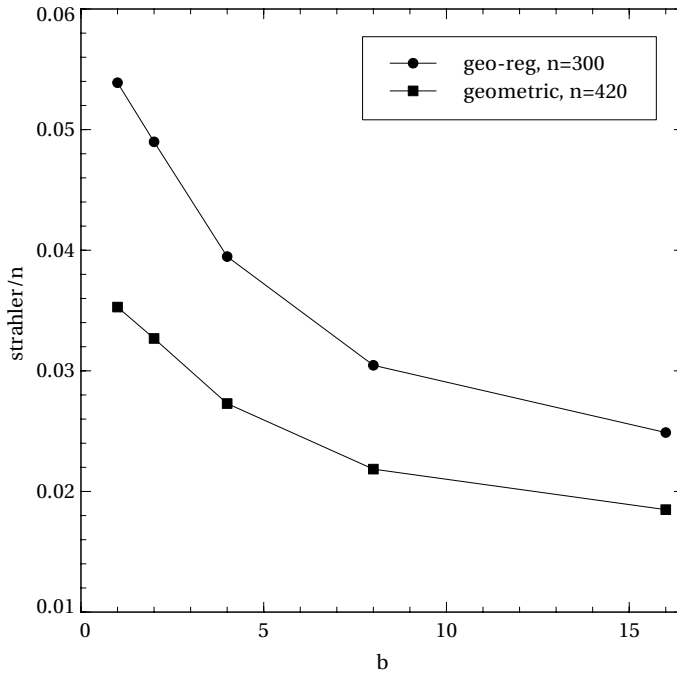


Figure 4. Strahler/variable ratio as a function of b , for big numbers of variables.

terms of hardness the geometric method for smaller values of b 's. The second row at the figure 3 shows the strahler/variable ratio as a function of the number of variables. It is interesting to see that it seems that this ratio tends to a fixed point as we increase the number of variables. Finally, figure 4 shows this potential fixed point as a function of b .

3. Conclusions

We have proposed a generalization of the uniform and the regular k -CNF random models, by generalizing the probability distribution used on the selection of variables to a geometric distribution with base b . We have experimentally located the phase transition point, for several values of the base b . We have also studied the problem hardness as a function of the number of variables and the base.

This will allow us to generate instances, at the phase transition point, of any given number of variables and *hardness* by adjusting the value of the parameter b . This is an important result since in order to do the same with the standard generators (uniform and regular random) we have to move to the under-constrained or over-constrained regions, where we find less interesting problems.

References

- [ABLM08] Carlos Ansótegui, María Luisa Bonet, Jordi Levy, and Felip Manyà. Measuring the hardness of sat instances. In *Proc. of the 23th AAAI Conference on Artificial Intelligence, AAAI'08*, 2008.

- [BDIS05] Yacine Boufkhad, Olivier Dubois, Yannet Interian, and Bart Selman. Regular random k -sat: Properties of balanced formulas. *J. Autom. Reasoning*, 35(1-3):181–200, 2005.
- [BKPS02] Paul Beame, Richard M. Karp, Toniann Pitassi, and Michael E. Saks. The efficiency of resolution and davis–putnam procedures. *SIAM J. Comput.*, 31(4):1048–1075, 2002.
- [BSW01] Eli Ben-Sasson and Avi Wigderson. Short proofs are narrow - resolution made simple. *J. ACM*, 48(2):149–169, 2001.
- [CA96] James M. Crawford and Larry D. Auton. Experimental results on the crossover point in random 3-SAT. *Artificial Intelligence*, 81(1-2):31–57, 1996.
- [CS88] Vasek Chvátal and Endre Szemerédi. Many hard examples for resolution. *J. ACM*, 35(4):759–768, 1988.
- [Dec03] Rina Dechter. *Constraint Processing*. Morgan Kaufmann, 2003.
- [JS96] Roberto J. Bayardo Jr. and Robert Schrag. Using csp look-back techniques to solve exceptionally hard sat instances. In Eugene C. Freuder, editor, *Proc. of the Second Int. Conf. on Principles and Practice of Constraint Programming, CP'96*, volume 1118 of *Lecture Notes in Computer Science*, pages 46–60. Springer, 1996.
- [KS99] Henry A. Kautz and Bart Selman. Unifying sat-based and graph-based planning. In Thomas Dean, editor, *Proc. of the 16th Int. Joint Conf. on Artificial Intelligence, IJCAI'99*, pages 318–325. Morgan Kaufmann, 1999.
- [KS03] Henry A. Kautz and Bart Selman. Ten challenges redux: Recent progress in propositional reasoning and search. In Francesca Rossi, editor, *Proc. of the 9th Int. Conf. on Principles and Practice of Constraint Programming, CP 2003*, volume 2833 of *Lecture Notes in Computer Science*, pages 1–18. Springer, 2003.
- [LA97] Chu Min Li and Anbulagan. Look-ahead versus look-back for satisfiability problems. In *Proc. of the 13th Int. Conf. on Principles and Practice of Constraint Programming, CP'07*, volume 4741 of *Lecture Notes in Computer Science*, pages 341–355. Springer, 1997.
- [MSL92] David G. Mitchell, Bart Selman, and Hector J. Levesque. Hard and easy distributions of sat problems. In *Proc. of the 10th National Conf. on Artificial Intelligence*, pages 459–465, 1992.
- [Sel00] Bart Selman. Satisfiability testing: Recent developments and challenge problems. In *Proc. of the 15th Annual IEEE Symposium on Logic in Computer Science, LICS'00*, page 178, 2000.
- [SKM97] Bart Selman, Henry A. Kautz, and David A. McAllester. Ten challenges in propositional reasoning and search. In *Proc. of the 15th Int. Joint Conf. on Artificial Intelligence, IJCAI'97*, pages 50–54, 1997.
- [WGS03] Ryan Williams, Carla P. Gomes, and Bart Selman. Backdoors to typical case complexity. In *Proc. of the 18th Int. Joint Conf. on Artificial Intelligence, IJCAI-03*, pages 1173–1178. Morgan Kaufmann, 2003.

Privacy in Distributed Meeting Scheduling¹

Ismel Brito and Pedro Meseguer²

IIIA, CSIC, Campus UAB, 08193 Bellaterra, Spain

Abstract. Meetings are an important vehicle for human interaction. The Meeting Scheduling problem (*MS*) considers several agents, each holding a personal calendar, and a number of meetings which have to be scheduled among subsets of agents. *MS* is a naturally distributed problem with a clear motivation to avoid centralization: agents desire to keep their personal calendars as private as possible during resolution. *MS* can be formulated as Distributed CSP, but due to the form of its constraints the PKC model does not bring any benefit here. We take entropy as a measure for privacy, and evaluate several distributed algorithms for *MS* according to efficiency and privacy loss. Experiments show interesting results with respect to the kind of tested algorithms.

Keywords. Distributed constraints, privacy, entropy.

1. Introduction

The Meeting Scheduling problem (*MS*) consists of a set of agents, each holding a personal calendar where previous appointments may appear, and a set of meetings among them. The goal is to determine *when* and *where* these meetings could occur [7]. This problem is naturally distributed because (i) each agent knows only his/her own personal calendar and (ii) agents usually desire to preserve their personal calendars during resolution. In a centralized approach, all agents must give their private information to a central server, which solves the problem and returns a solution. This causes a high privacy loss, because each agent has to give his/her personal calendar to the server. In a distributed approach, to find a solution some information of the personal calendars has to be revealed, but not all of them as in the centralized case.

A natural formulation of *MS* is as Distributed Constraint Satisfaction (*DisCSP*) with privacy requirements. To enforce privacy in *DisCSP*, two main approaches have been explored. One considers the use of cryptographic techniques [10], which causes significant overhead in solving. Alternatively, other authors try to enforce privacy by different search strategies [7,6]. In this paper, we follow this line. It is worth noting that the *partially-known constraints* (PKC) approach [2,4], developed to enforce privacy in *DisCSP*, it does not offer benefits for this problem because an equality constraint cannot

¹This work is partially supported by the project TIN2006-15387-C03-01 and by the Generalitat de Catalunya grant 2005-SGR-00093.

²Corresponding Author: IIIA CSIC, Campus UAB, 08193 Bellaterra, Spain; email: pedro@iiia.csic.es

be divided in two private parts (one for each agent of a binary constraint), and this problem is modelled using equality constraints. We analyze three distributed approaches for *MS*, which can be seen as representative algorithms of the state-of-the-art in distributed constraint solving. One is the simple *RR* algorithm [7], developed for this problem. The other two are generic *DisCSP* algorithms: the synchronous *SCBJ* and the asynchronous *ABT*. We compare them experimentally, with especial emphasis on the privacy level they can reach. Privacy is measured using entropy from information theory, as done in [4].

This paper is structured as follows. In section 2, we give a formal definition of the *MS* problem, providing a *DisCSP* encoding. In section 3, we present entropy as a measure for privacy and we discuss its applicability for the considered solving approaches. In section 4, we discuss the idea of using lies to enforce privacy. In section 5, we compare empirically the considered algorithms, and we extract some conclusions in section 6.

2. The Meeting Scheduling Problem

The *MS* problem [7] involves a set of agents and a set of meetings among them. The goal is to decide *when* and *where* these meetings could be scheduled. Formally, a *MS* is defined as (A, M, S, P) , where $A = \{a_1, a_2, \dots, a_n\}$ is a set of n agents; $M = \{m_1, m_2, \dots, m_q\}$ is a set of q meetings; $att(m_k)$ is the set of m_k attendees; $S = \{s_1, s_2, \dots, s_r\}$ is the set of r slots in any agent's calendar; $P = \{p_1, p_2, \dots, p_o\}$ is the set of places where meetings can occur. We also define the set of meetings where agent a_i is involved as $M_i = \{m_j | a_i \in att(m_j)\}$, the common meetings between agents a_i and a_j as $M_{ij} = M_i \cap M_j$, and the set of agents connected with a_i as $A_i = \cup_{m_j \in M_i} att(m_j)$. Initially, agents may have several slots reserved for already filled planning in their calendars. A solution must assign a time and a place to each meeting, such that the following constraints are satisfied (i) a meeting attendees must agree *where* and *when* it will occur; (ii) m_i and m_j cannot be held at same time if they have one common attendee; (iii) each attendee a_i of meeting m_j must have enough time to travel from the place where he/she is before the meeting to the place where the meeting m_j will be; human agents need enough time to travel to the place where their next meeting will occur.

MS is a truly distributed benchmark, in which each attendee may desire to keep the already planned meetings in his/her calendar private. This problem is very suitable to be treated by distributed techniques, trying to provide more autonomy to each agent while enforcing privacy. With this purpose, we formulate the Distributed Meeting Scheduling (*DisMS*) problem, which can be seen as a Distributed Constraint Satisfaction problem (*DisCSP*). One agent in *DisMS* corresponds exactly with an agent in *DisCSP*. Each agent includes one variable per meeting in which it participates. The variable domain enumerates all the possible alternatives of *where* and *when* the meeting may occur. Each domain has $o \times r$ values, where o is the number of places where meetings can be scheduled and r represents the number of slots in agents' calendars. There are two types of constraints: equality and difference constraints. There is a binary equality constraint between each pair of variables of different agents that corresponds to the same meeting. There is an all-different constraint between all variables that belong to the same agent.

3. Privacy in *DisMS* Algorithms

To solve a *DisMS* instance, agents exchange messages looking for a solution. During this process, agents leak some information about their personal calendars. Privacy loss is concerned with the amount of information that agents reveal to other agents. In the *DisCSP* formulation for *DisMS*, variable domains represent the availability of agents to hold a meeting at a given time and place, which is the information that agents desire to hide from other agents. In that sense, measuring the privacy loss of a *DisMS* modeled as *DisCSP* is the same as measuring the privacy loss of variable domains.

Following [4], we use entropy from information theory as a quantitative measure for privacy. The entropy of a random variable Z taking values in the discrete set S is,

$$H(Z) = - \sum_{i \in S} p_i \log_2 p_i$$

where p_i is the probability that Z takes the value i . $H(Z)$ measures the amount of missing information about the possible values of the random variable Z [9,5]. If only one value k is possible for that variable, there is no uncertainty about the state of Z , and $H(Z) = 0$. Given a *DisCSP*, let us consider agent a_j . The rest of the problem can be considered as a random variable, with a discrete set of possible states S . Applying the concept of information entropy, we define the entropy H_j associated with agent a_j and the entropy H associated with the whole problem as,

$$H_j = - \sum_{i \in S} p_i \log_2 p_i \quad H = \sum_{j \in A} H_j$$

Solving can be seen as an entropy-decrement process. Initially, agents know nothing about other agents. If a solution is reached after distributed search, agent a_i has no uncertainty about the values of its meetings (that also appear in other agents) so its entropy decrements. In addition, some information is leaked during search, which contributes to this decrement. We take the entropy decrement in solving as a measure of privacy loss, and this allows us to compare different algorithms with respect to privacy. We consider three particular states: *init*, it is the initial state, where agents know little about others; *sol*, it is when a solution has been found after solving; we are interested in the current states of the agents' calendars; *end*, it is the state after the solving process, no matter whether a solution has been found or not; here, we are interested in the initial state of agents' calendars. Assessing entropy of *sol* state, we evaluate how much of the reached solution is known by other agents. Assessing entropy of the *end* state, we evaluate how much information about the initial state has leaked in the solving process. In the following, we present *RR*, *SCBJ* and *ABT*, and their corresponding entropies for these states.

3.1. The *RR* Algorithm

RR was introduced in [7] to solve *DisMS*. This algorithm is based on a very simple communication protocol: agent a_i considers one of its meetings m_j , and proposes a time/place to the other agents in $att(m_j)$. These agents answer a_i with acceptance/rejection, depending whether the proposed time/place is acceptable or not accord-

ing to their calendars. In both cases, another agent takes control and proposes (i) a new time/place for m_j if it was not accepted in the previous round, or (ii) a time/place proposal for one of its meetings, if m_j was accepted. If an agent finds that no value exists for a particular meeting, backtracking occurs, and the latest decision taken is reconsidered. The process continues until finding a solution for every meeting (a whole solution for the *DisMS* problem), or when the first agent in the ordering performs backtracking (meaning that no global solution exists). Agent activation follows a Round Robin strategy.

RR agents exchange six types of messages: **pro**, **ok?**, **gd**, **ngd**, **sol**, **stp**. When a_i receives a **pro** message, this causes a_i to become the proposing agent. It considers one of its meetings m_j with no assigned value: a_i chooses a time/place for m_j and ask for agreement to other agents in $att(m_j)$ via **ok?** messages. When a_k receives an **ok?** message, it checks if the received proposal is valid with respect to previously scheduled appointments in its calendar. If so, a_k sends a **gd** message to a_i announcing that it accepts the proposal. Otherwise, a_k sends a **ngd** message to a_i meaning rejection. If a_i exhausts all its values for m_j without reaching agreement, it performs backtracking to the previous agent in the round-robin. Messages **sol** and **stp** announce to agents that a solution has been found or the problem is unsolvable, respectively.

Before search starts, the entropy of agent a_i is

$$H_i(init) = - \sum_{a_k \in A, k \neq i} \sum_{l=1}^d p_l \log_2 p_l = \sum_{a_k \in A, k \neq i} \log_2 d$$

where we assume that meetings have a common domain of size $d = r \times o$, whose values have the same probability $p_l = \frac{1}{d} \forall l$. In the *sol* state, when a solution is found, agent a_i knows the values of meetings in common with other agents, so its entropy is,

$$H_i(sol) = \sum_{a_k \in A_i, k \neq i} \log_2(d - |M_{ik}|) + \sum_{a_k \in A - A_i} \log_2 d$$

that is, the contribution of agents connected with a_i has decreased because a_i knows some entries their calendars, while the contribution of unconnected agents remains the same. Meetings in common with a_i is the minimum information that a_i will have in the *sol* state. Therefore, the previous expression is optimal for *DisMS*. The information leaked during the solving process has little influence in the entropy associated with the *sol* state, because an entry in a_i 's calendar that is revealed as free in a round for a particular meeting m_j , it could be occupied by another meeting m_k involving a different set of attendees. However, this information is very relevant to dig into the initial state of other agents' calendars. During the solving process, each time a_i reveals that a slot is free (because it proposes that slot or because it accepts a proposal including that slot by another agent), it reveals that it was free at the beginning. This accumulates during search, and when search ends a_i knows some entries in the initial calendar of a_k . In the *end* state the entropy is,

$$H_i(end) = \sum_{a_k \in A_i, k \neq i} \log_2(d - free_k^i) + \sum_{a_k \in A - A_i} \log_2 d$$

where $free_k^i$ is the number of different entries of a_k calendar that a_i knows were free at the beginning. It has been evaluated experimentally in section 5.

3.2. SCBJ

Synchronous Conflict-based Backjumping algorithm (*SCBJ*) is the synchronous distributed version of the well-known *CBJ* algorithm in the centralized case. *SCBJ* assigns variables sequentially, one by one. It sends to the next variable to assign the whole partial solution, that contains all assignments of previous variables. This variable tries a value and checks if this value is consistent with previous assignments. If so, that variable remains assigned and the new partial solution (that includes this new assignment) is sent to the next variable. If this value is inconsistent, a new value is tried. If the variable exhausts all its values, backjumping is performed to the last previous variable responsible for the conflict. Agents implement the described algorithm by exchanging assignments and no-goods through **ok?** and **ngd** messages, respectively. From the point of view of *DisMS*, agents accept or reject the proposals made by other agents. **ok?** messages are used for the agents to send proposals regarding time/place that are acceptable for a meeting. Contrary to what happens in *RR*, **ngd** messages only mean that someone has rejected the proposal, but the agent who has done such is not easily discovered. It is important to note that *SCBJ* loses some possible privacy in the sense that as the agents send **ok?** messages down the line, each agent knows that all the previous agents have accepted this proposal.

The entropy associated with the *init* state is as in the *RR* case. In the *sol* state, a_i knows the value of every meeting in M_i , and the values of meetings scheduled prior its own meetings (there values were included in the last partial solution that reached a_i). So

$$H_i(sol) = \sum_{a_k \in A_i, k \neq i} \log_2(d - |M_{ik}|) + \sum_{a_k \in A - A_i} \log_2(d - known_k^i)$$

where $known_k^i$ is the number of meetings of a_k unconnected with a_i , whose time/place are known by a_i because they were scheduled before its own meetings. It is easy to see that *SCBJ* is less private than *RR* comparing their expressions of $H_i(sol)$. The first term is equal, but the second term is lower for *SCBJ* than for *RR* (it may be equal for some i , but for others has to be necessarily lower). This entropy decrement of *SCBJ* is due to the fact that a_i , besides knowing meetings in M_i (something shared with *RR*), it also knows the times/places of some other meetings where a_i does not participate. This is a serious drawback of this algorithm regarding privacy.

Regarding the initial state of calendars, a similar analysis to the one done for *RR* applies here. Each time a_i receives a partial solution with proposals, a free slot in the calendar of the proposing agent is revealed, and this information accumulates during search. At the end, the entropy associated with initial domains is,

$$H_i(end) = \sum_{a_k \in A, k \neq i} \log_2(d - free_k^i)$$

where $free_k^i$ is the number of free slots that a_i detects in a_k . It has been evaluated experimentally in section 5.

3.3. ABT

Asynchronous Backtracking (*ABT*) [11] is an asynchronous algorithm that solves *DisCSPs*. Agents in *ABT* assign their variables asynchronously and concurrently. *ABT*

computes a solution (or detects that no solution exists) in finite time; it is correct and complete. *ABT* requires constraints to be directed. A constraint causes a directed link between two constrained agents. To make the network cycle-free, there is a total order among agents that corresponds to the directed links. Agent a_i has higher priority than agent a_j if a_i appears before a_j in the total order. Each *ABT* agent keeps its own agent view and nogood list. The agent view of a_i is the set of values that it believes to be assigned to agents connected to a_i by incoming links. The nogood list keeps the nogoods received by a_i as justifications of inconsistent values.

ABT agents exchange four types of messages: **ok?**, **ngd**, **addl**, **stp**. *ABT* starts by each agent assigning their variables, and sending these assignments to connected agents with lower priority via **ok?** messages. When an agent receives an assignment, it updates its agent view, removes inconsistent nogoods and checks the consistency of its current assignment with the updated agent view. If the current assignment of one of its variables is inconsistent, a new consistent value is searched. If no consistent value can be found, a **ngd** message is generated. When receiving a nogood, it is accepted if it is consistent with the agent view of a_i . Otherwise, it is discarded as obsolete. An accepted nogood is used to update the nogood list. It makes a_i search for a new consistent value of the considered variable, since the received nogood is a justification that forbids its current value. When an agent cannot find any value consistent with its agent view, either because of the original constraints or because of the received nogoods, new nogoods are generated from its agent view and each one sent to the closest agent involved in it, causing backtracking. In addition, if a_i receives a **ngd** message mentioning an agent a_j not connected with a_i , a message **addl** is sent from a_i to a_j , asking for a new directed link, that will be permanent from this point on. The message **stp** means that no solution exists. When a solution is found, this is detected by quiescence in the network.

The entropy associated with the *init* state is equal to the *RR* case. In the *sol* state, a_i knows the value of every meeting in M_i . It also knows the values of meetings corresponding to variables initially unconnected with a_i but later connected by added links,

$$H_i(sol) = \sum_{a_k \in A_i, k \neq i} \log_2(d - |M_{ik}|) + \sum_{a_k \in A - A_i} \log_2(d - link_k^i)$$

where $link_k^i$ is the number of meetings of a_k initially unconnected with a_i , whose time/place are known by a_i because they were connected by its new links during search. It has been evaluated experimentally in section 5.

It is easy to see that *ABT* is less private than *RR* comparing their expressions of $H_i(sol)$. The first term is equal, but the second term of *ABT* is lower than or equal to the second term of *RR*. This entropy decrement of *ABT* is due to the fact that a_i , in addition knowing meetings in M_i (something shared by *RR*), it also knows the times/places of some other meetings where a_i does not participate. It is worth noting here that there is a version of *ABT* that does not add new links during search [1]. For this version, called *ABT_{not}*, its privacy is equal to the obtained by *RR*, considering the *sol* state. With respect to *SCBJ*, no analytical result can be extracted: both share the first term, but their second terms are incomparable. Regarding initial domains, a similar analysis to the one done for *RR* and *SCBJ* applies here. Each time a_i receives a proposal, a free slot in the calendar of the proposing agent is revealed. This information accumulates during search. At the end, the entropy associated with initial domains is,

$$H_i(end) = \sum_{a_k \in A, k \neq i} \log_2(d - free_k^i)$$

where $free_k^i$ is the number of free slots that a_i detects in a_k (a_k has to be higher than a_i in the total order). It has been evaluated experimentally in section 5.

4. Allowing Lies

A simple way to enforce privacy is to allow agents to lie: to declare they have assigned some value while this is not really true [3]. Formally, if variable i has m values $D_i = \{v_1, v_2, \dots, v_m\}$, allowing lies means enlarging i 's domain $D'_i = \{v_1, v_2, \dots, v_m, v_{m+1}, \dots, v_{m+t}\}$ with t extra values. The first m values are *true values*, while the rest are *false values*. If an agent assigns one of its variables with a false value and informs other agents, the agent is sending a lie. We assume that agents may lie when informing other agents, but they always say the truth when answering other agents.

If lies are allowed, an agent receiving a message cannot be sure whether it contains true or false information. The accumulation of true information during search is more difficult than when lies were not allowed. Lies are a way to conceal true information by adding some kind of *noise* that makes more difficult any inference on the data revealed during search. But this has a price in efficiency. Since the search space is artificially enlarged with false values, its traversal requires more effort. So using lies increases the privacy level achieved by solving algorithms but decreases their efficiency.

A solution reported by the algorithm must be a true solution, and it should not be based on a lie. If an agent has sent a lie, it has to change and say the truth in finite time afterwards. This is a sufficient condition to guarantee correctness in asynchronous algorithms (assuming that the time to say the truth is shorter than the quiescence time of the network) [3]. The inclusion of the lie strategy in synchronous algorithms presents some difficulties, as we see next. *RR* cannot use lies. If the first proposing agent sends a lie that is accepted by all other agents, this agent has no way to retract its lie. The control is passed to another agent, which if it does not see any conflict it passes control to another agent, etc. until all agents have been activated following the round-robin strategy. If lies are allowed, the *RR* algorithm might generate incorrect solutions. *SCBJ* cannot use lies, for similar reasons. An agent proposes when the partial solution reaches it. After assigning its variables, the new partial solution departs from it, and the agent has no longer control of that solution. If the agent introduces a lie in the partial solution, it may be the case that it could not retract the lie in the future, causing incorrect solutions. So using lies causes *SCBJ* to lose its correctness. *ABT* can use lies. An agent may change its value asynchronously. If an agent has sent a lie, it can send another true value in finite time afterwards. Sending lies is not a problem for *ABT*-based algorithms [3].

The use of lies in *ABT* causes some changes in the information revealed during search. Each time a_i receives a proposal, it is unsure whether it contains a true value (a free slot in the proposing agent's calendar) or a false value (an occupied slot taken as free). Even if the proposing agent changes afterwards, it is unclear whether this is caused because the previous proposal was false or because it has been forced to change by search. Since there is uncertainty about this, no true information can be accumulated during search. At the end and only when there is a solution, the entropy associated with initial domains is equal to the entropy of the *sol* state.

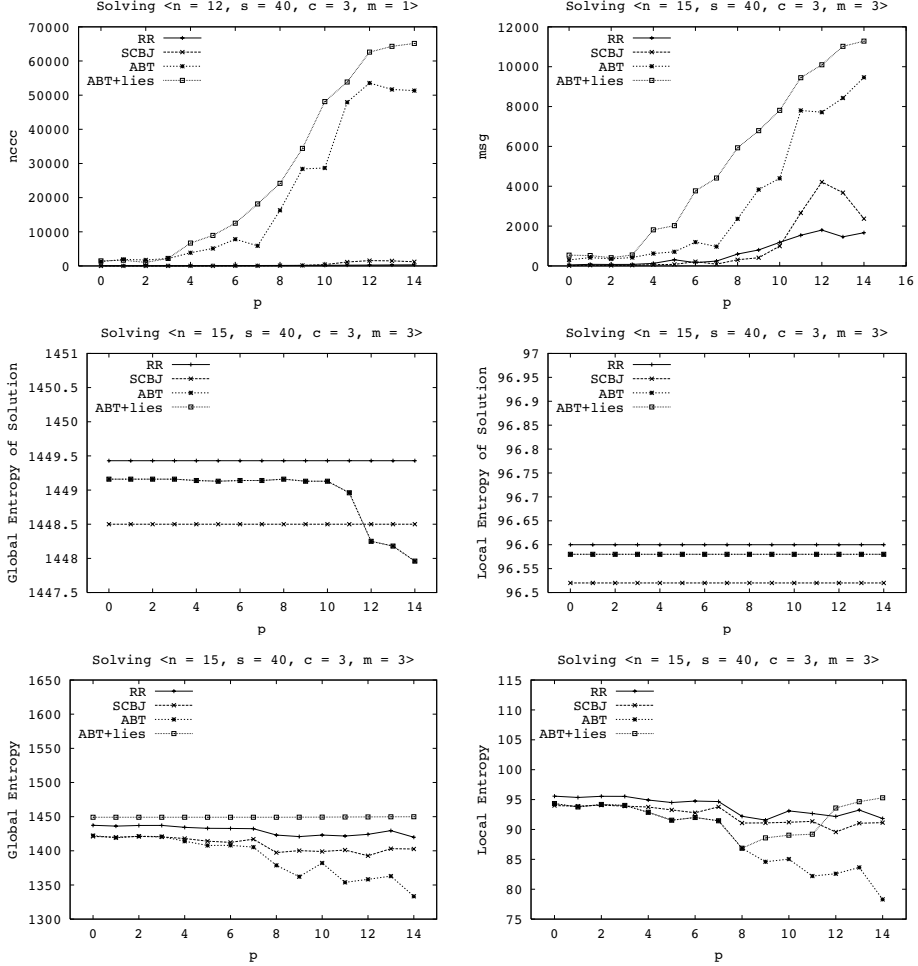


Figure 1. Experimental results of *RR*, *SCBJ*, *ABT* and *ABT_{lies}* on random *DisMS* instances. Top row: non-concurrent constraint checks and number of exchanged messages. Middle row: global and local entropy of *sol* state. Bottom row: global and local entropy of *end* state.

5. Experimental Results

We evaluate *RR*, *SCBJ*, *ABT* and *ABT_{lies}* (the *ABT* version where lies are allowed), on random meeting scheduling instances. We compare these algorithms using three measures: computation effort, communication cost and privacy loss. We measure computation effort using the number of non concurrent constraint checks (*nccc*) [8], communication cost as the number of messages exchanged (*msg*) and privacy loss in terms of entropy in *sol* and *end* states (entropy of the *init* state depends on the instance only).

In *SCBJ* and *ABT*, lower priority agents tend to work more than higher priority ones, which causes them to reveal more information. This generates an uneven distribution of privacy loss among agents as search progresses. Because of this, we provide values of global and local entropy for each algorithm. The global entropy, H , is the sum of agents' individual entropies. The local entropy is the minimum of agents' entropies, and it aims

at assessing the amount of information that the best informed agent infers from others' calendars. In both cases, higher values of entropy mean higher privacy.

The experimental setting considers *DisMS* instances with three meetings to schedule. Each instance is composed of 15 people, 5 days with 8 time slots per day and 3 meeting places. This gives $5 \cdot 8 \cdot 3 = 120$ possible values in each variable's domain. Meetings and time slots are both one hour long. The time required for travel among the three cities is 1 hour, 1 hour and 2 hours. *DisMS* instances are generated by randomly establishing p predefined meetings in each agent's calendar. Parameter p varies from 0 to 14.

In *RR*, we count one constraint check each time that an agent checks if a meeting can occur at a certain time/place. In all algorithms, each time an agent has to make a proposal, it chooses a time/place at random. Agents in *ABT* process messages by packets instead of processing one by one and implement the strategy of selecting the best nogood [1]. In *ABT_{lies}*, when an agent searches for a new consistent value, it randomly decides if it chooses a true or false value, so sending a truth value or a lie is equally probable.

Figure 1 shows the average results of 20 *DisMS* instances for each value of p . Regarding computation effort (*nccc*), we observe that the synchronous algorithms need fewer *nccc* than asynchronous ones, with *RR* showing the best performance. This phenomenon can be explained by analyzing how agents reject proposals. In *RR* the proposing agent broadcasts its meeting proposal to all agents sharing a constraint with it. When those agents receive the proposal, they determine if it is valid or not according to their own calendars. The proposal checking process can be concurrently executed by informed agents. In terms of *nccc*, this means that the total cost of all checking processes executed by informed agents is equal to the cost of one checking process (1 *nccc*). In *SBCJ* the checking process for a proposal involves more agents because a proposal made by agent a_i will reach a_j passing through intermediate agents (the number of intermediate agents depends on the ordering among agents) and further includes other proposals made by these intermediate agents. Finally, if the proposal made by a_i is found inconsistent by a_j , this will represent a substantial number of *nccc*. Contrary to the previous algorithms, *ABT* agents send proposals asynchronously, while knowing the proposals of higher priority agents. Since consistency among constraining proposals occurs when they have the same value, and values for the proposals are randomly selected, consistency occurs only after several trials. When considering lies, *ABT_{lies}* exhibits worse performance than pure *ABT* as expected - with lies performance deteriorates. Regarding communication cost (*msg*), the same ordering among algorithms occurs, for similar reasons.

As discussed in Section 3, we have used entropy at *sol* and *end* states as quantitative metrics of privacy of the four algorithms. The middle row of Figure 1 corresponds to the global and local entropies at the *sol* state. In *RR*, we see that both metrics of entropy are constant because they are independent as far as search effects go. They also show a constant behavior for *SCBJ* because a static variable ordering has been used. The global entropy of *ABT* decreases when the number of predefined meetings increases. This is because finding a solution is harder when p increases, making agents add new links and thus decreasing privacy. The entropy of the best informed *ABT* agent (given by the local entropy) remains practically constant with respect to p . These points corroborate what was theoretically proven in Section 3: *RR* has higher entropy in *sol* than *SCBJ* and *ABT*. The bottom row of Figure 1 shows the global and local entropies at the *end* state of the four algorithms. Interestingly, the algorithm offering the highest global privacy is *ABT_{lies}*, while *ABT* is the algorithm offering the lowest global privacy. Allowing lies is a simple

way to achieve high privacy (at the extra cost of more computation and communication costs). The entropy of the best informed agent tends to decrease as p increases, keeping the relative order RR , $SCBJ$, ABT . At some point, ABT_{lies} separates from ABT and becomes the most private among them. This is due to the presence of unsolvable instances in the right part of the plot, which have the minimum contribution to the entropy of any agent of the system.

6. Conclusions

We have analyzed privacy loss of several distributed algorithms for *DisMS*, a naturally distributed problem of practical interest. We have used entropy as a quantitative measure for privacy, considering how much information leaks from the final solution and from initial calendars during search. We have also discussed the use of lies during resolution, to enforce privacy. Our experimental results show that the two synchronous approaches outperform the asynchronous ones in computation effort and communication cost. Regarding privacy, the picture is a bit more complex. Considering privacy of the final solution, the synchronous RR algorithm shows the best privacy. Considering privacy of initial calendars (due to information leaked during search), ABT_{lies} reaches the highest global privacy. All in all, the simple RR offers a very high privacy with unbeaten performance.

References

- [1] C. Bessière, A. Maestre, I. Brito, P. Meseguer. Asynchronous Backtracking without Adding Links: a New Member to ABT Family. *Artificial Intelligence*, **161**(1–2), 7–24, 2005.
- [2] I. Brito and P. Meseguer. Distributed Forward Checking. *Proc. of 8th CP*, 801–806, 2003.
- [3] I. Brito and P. Meseguer. Distributed Forward Checking May lie for Privacy. *Recent Advances in Constraints*, LNAI 4651, Ed. F. Azevedo, 93–107, 2007.
- [4] I. Brito and A. Meisels and P. Meseguer and R. Zivan. Distributed Constraint satisfaction with Partially Known Constraints. *Constraints*, accepted for publication. 2008.
- [5] T. M. Cover and J. A. Thomas. *Elements of Information Theory*, Wiley-Interscience, 2nd edition, 2006.
- [6] M. S. Franzin and F. Rossi and E. C. Freuder and R. Wallace. Multi-Agent Constraint Systems with Preferences: Efficiency, Solution Quality, and Privacy Loss. *Computational Intelligence*, **20**, 264–286, 2004.
- [7] Freuder E.C., Minca M., Wallace R.J. Privacy/efficiency trade-offs in distributed meeting scheduling by constraint-based agents. *Proc. of DCR Workshop at IJCAI-01*, 63–71, USA, 2001.
- [8] Meisels A., Kaplansky E., Razgon I., Zivan R. Comparing performance of distributed constraint processing algorithms. *Proc. of DCR Workshop at AAMAS-02*, 86–93, Italy, 2002.
- [9] C. E. Shannon. *The Mathematical Theory of Communication*, University of Illinois Press, 1963.
- [10] M. C. Silaghi. Meeting Scheduling Guaranteeing $n/2$ -Privacy and Resistant to Statistical Analysis (Applicable to any DisCSP). *Proc. of the 3th Conference on Web Intelligence*, 711–715, 2004.
- [11] M. Yokoo, E. H. Durfee, T. Ishida, K. Kuwabara. The Distributed Constraint Satisfaction Problem: Formalization and Algorithms. *IEEE Trans. Knowledge and Data Engineering*, **10**, 673–685, 1998.

Solving the Response Time Variability Problem by means of Multi-start and GRASP metaheuristics¹

Albert COROMINAS, Alberto GARCÍA-VILLORIA², Rafael PASTOR
*Institute of Industrial and Control Engineering (IOC), Technical University of
 Catalonia (UPC), Barcelona, Spain*

Abstract. The Response Time Variability Problem (RTVP) is an NP-hard scheduling optimization problem that has recently appeared in the literature. This problem has a wide range of real-life applications in, for example, manufacturing, hard real-time systems, operating systems and network environments. The RTVP occurs whenever models, clients or jobs need to be sequenced to minimize variability in the time between the instants at which they receive the necessary resources. The RTVP has been already solved in the literature with a multi-start and a GRASP algorithm. We propose an improved multi-start and an improved GRASP algorithm to solve the RTVP. The computational experiment shows that, on average, the results obtained with our proposed algorithms improve on the best obtained results to date.

Keywords. response time variability, regular sequences, scheduling, multi-start metaheuristic, grasp

Introduction

The Response Time Variability Problem (RTVP) is a scheduling problem that has recently been formalized in [1]. The RTVP occurs whenever products, clients or jobs need to be sequenced so as to minimize variability in the time between the instants at which they receive the necessary resources. Although this optimization problem is easy to formulate, it is very difficult to solve optimally (it is NP-hard [1]).

The RTVP has a broad range of real-life applications. For example, it can be used to regularly sequencing models in the automobile industry [2], to allocating resources in computer multi-threaded systems and network servers [3], to broadcasting video and sound data frames of applications over asynchronous transfer mode networks [4], in the periodic machine maintenance problem when the distances between consecutive services of the same machine are equal [5] and in the collection of waste [6].

One of the first problems in which has appeared the importance of sequencing *regularly* is the sequencing on the mixed-model assembly production lines at Toyota Motor Corporation under the just-in-time (JIT) production system. One of the most

¹ Sponsored by the Spanish Ministry of Education and Science's project DPI2007-61905; co-funded by the FEDER.

² Corresponding author: Alberto García-Villoria, IOC – Institute of Industrial and Control Engineering, Av. Diagonal 647 (Edif. ETSEIB), 11th floor, 08028 Barcelona, Spain; e-mail: alberto.garcia-villoria@upc.edu

important JIT objectives is to get rid of all kinds of waste and inefficiency and, according to Toyota, the main waste is due to the stocks. To reduce the stock, JIT production systems require to producing only the necessary models in the necessary quantities at the necessary time. To achieve this, one main goal, as Monden says [2], is scheduling the units to be produced to keep constant consumption rates of the components involved in the production process. Miltenburg [7] deals with this scheduling problem and he assumes that models require approximately the same number and mix of parts. Thus, he considers only the demand rates for the models. In our experience with practitioners of manufacturing industries, we noticed that they usually refer to a good mixed-model sequence in terms of having distances between the units for the same model as regular as possible. Therefore, the metric used in the RTVP reflects the way in which practitioners refer to a desirable regular sequence.

In [1], a mixed integer lineal programming (MILP) model to solve the RTVP has been proposed. The previous MILP model has been improved in [8], but the practical limit to obtain optimal solutions is 40 units to be scheduled. Thus, the use of heuristic or metaheuristic methods for solving real-life RTVP instances is justified. In [1], five greedy heuristic algorithms have been proposed. Seven metaheuristic algorithms -one multi-start, one GRASP (Greedy Randomized Adaptive Search Procedure) and four PSO (Particle Swarm Optimization) algorithms- have been proposed in [9]. Finally, eleven PSO metaheuristic algorithms were used to solve the RTVP in [10].

The general scheme of the multi-start metaheuristic consists of two phases. In the first phase an initial solution is generated. Then, the second phase improves the obtained initial solution. These two phases are iteratively applied until a stop condition is reached. The GRASP metaheuristic can be considered a variant of the multi-start metaheuristic in which the initial solutions are obtained using direct randomness. They are generated by means of a greedy strategy in which random steps are added and the choice of the elements to be included in the solution is adaptive.

This paper is an extension of the work initialized in [9]. The new research done with the PSO metaheuristic was reported in [10] and the PSO algorithm called *DPSOpoi-c_pdyn* by the authors is the best algorithm to date for solving the RTVP. In this paper we propose an improved multi-start algorithm and an improved GRASP algorithm. On average, the proposed algorithms improve strongly on previous results.

The rest of this paper is organized as follows. Section 1 presents a formal definition of the RTVP. Section 2 explains the existing multi-start and GRASP for solving the RTVP and proposes two new improved multi-start and GRASP algorithms. Section 3 provides the computational experiment and the comparison with the best algorithm to solve the RTVP (*DPSOpoi-c_pdyn*) and the existing multi-start and GRASP algorithms. Finally, some conclusions are given in Section 4.

1. Response Time Variability Problem (RTVP)

The aim of the Response Time Variability Problem (RTVP) is to minimize the variability of the distances between any two consecutive units of the same model in the sequence.

The RTVP is stated as follows. Let n be the number of models, d_i the number of units of the model i ($i = 1, \dots, n$) to be scheduled and D the total number of units

($D = \sum_{i=1}^n d_i$). Let s be a solution of an instance of the RTVP. It consists in a circular sequence of units ($s = s_1 s_2 \dots s_D$), where s_j is the unit sequenced in position j of sequence s . For all model i in which $d_i \geq 2$, let t_k^i be the distance between the positions in which the units $k+1$ and k of the model i are found (where the distance between two consecutive positions is considered equal to 1). Since the sequence is circular, position 1 comes immediately after position D ; therefore, $t_{d_i}^i$ is the distance between the first unit of the model i in a cycle and the last unit of the same model in the preceding cycle. Let \bar{t}_i be the average distance between two consecutive units of the model i ($\bar{t}_i = D/d_i$). For all model i in which $d_i = 1$, t_1^i is equal to \bar{t}_i . The objective is to minimize the metric Response Time Variability (RTV), which is defined by the following expression: $RTV = \sum_{i=1}^n \sum_{k=1}^{d_i} (t_k^i - \bar{t}_i)^2$.

For example, let $n = 3$, $d_A = 2$, $d_B = 2$ and $d_C = 4$; thus, $D = 8$, $\bar{t}_A = 4$, $\bar{t}_B = 4$ and $\bar{t}_C = 2$. Any sequence such that contains exactly d_i times the symbol i ($\forall i$) is a feasible solution. For example, the sequence (C, A, C, B, C, B, A, C) is a solution, where

$$RTV = ((5-4)^2 + (3-4)^2) + ((2-4)^2 + (6-4)^2) + ((2-2)^2 + (2-2)^2 + (3-2)^2 + (1-2)^2) = 12.$$

As explained in the introduction, the best RTVP results recorded to date were obtained by using a PSO algorithm called *DPSOpoi-c_pdyn* [10]. PSO is a population metaheuristic algorithm based on the social behaviour of flocks of birds when they search for food. The population or swarm is composed of particles (birds), whose attributes are an n -dimensional real point (which represents a feasible solution) and a velocity (the movement of the point in the n -dimensional real space). The velocity of a particle is typically a combination of three types of velocities: 1) the inertia velocity (i.e., the previous velocity of the particle); 2) the velocity to the best point found by the particle; and 3) the velocity to the best point found by the swarm. These components of the particles are modified iteratively by the algorithm as it searches for an optimal solution. Although the PSO algorithm was originally designed for n -dimensional real spaces, *DPSOpoi-c_pdyn* is adapted to work with a sequence that represents the solution. Moreover, *DPSOpoi-c_pdyn* introduces random modifications to the points of the particles with a frequency that changes dynamically according to the homogeneity of the swarm (for more details, see [10]).

2. The multi-start and GRASP algorithms

2.1. The multi-start algorithm

The multi-start metaheuristic is a general scheme that consists of two phases. The first phase obtains an initial solution and the second phase improves the obtained initial solution. These two phases are applied iteratively until a stop condition is reached. This scheme has been first used at the beginning of 80's [11]. The generation of the initial

solution, how to improve them and the stop condition can be very simple or very sophisticated. The combination of these elements gives a wide variety of multi-start methods. For a good review of multi-start methods, see [12] and [13].

The multi-start algorithm proposed in [9] for solving the RTVP is based on generating, at each iteration, a random initial solution and on improving it by means of a local search procedure. The algorithm stops after it has run for a preset time. Random solutions are generated as follows. For each position, a model to be sequenced is randomly chosen. The probability of each model is equal to the number of units of this model that remain to be sequenced divided by the total number of units that remain to be sequenced. The local search procedure used is applied as follows. A local search is performed iteratively in a neighbourhood that is generated by interchanging each pair of two consecutive units of the sequence that represents the current solution; the best solution in the neighbourhood is chosen; the optimization ends when no neighbouring solution is better than the current solution.

If the quality of the initial solutions is low, the computing time required by the local search to find the local optimum is increased. For big RTVP instances, few iterations may be done because of the available execution time. An easy and fast way to obtain better initial solutions without giving up the simplicity of the multi-start algorithm could be generating, at each iteration, P random solutions and get as the initial solution the best of them, that is, applying the local search only for the best solution of the P random solutions. In this paper we propose a parametric multi-start algorithm to solve the RTVP that has one parameter: the number of random solutions generated at each iteration (P). Figure 1 shows the pseudocode of our algorithm.

Figure 1. Pseudocode of the proposed multi-start algorithm

1. Set the value of the parameter P
2. Let the best solution found \bar{X} initially be void
3. Let the RTV value of the best solution found be $\bar{Z} = \infty$
4. While execution time is not reached do:
 5. Generate P random solutions
 6. Let X the best solution generated at step 5
 7. Apply the local optimization to X and get X^{opt}
 8. If $RTV(X^{opt}) < \bar{Z}$, then $\bar{X} = X^{opt}$ and $\bar{Z} = RTV(X^{opt})$
9. End While
10. Return \bar{X}

As it has been mentioned, when the execution time of the algorithm is reached, the algorithm is immediately stopped (that is, the current local optimization is also stopped).

2.2. The GRASP algorithm

The GRASP metaheuristic was designed in 1989 by Feo and Resende [14] and can be considered as a multi-start variant. However, the generation of the initial solution is performed by means of a greedy strategy in which random steps are added and the choice of the elements to be included in the solution is adaptive.

The random step in the GRASP proposed in [9] consists of selecting the next model to be sequenced from a set called candidate list; the probability of each candidate model is proportional to the value of an associated index. The index used in [9] is the Webster index, which is evaluated as follows. Let x_{ik} be the number of units of model i that have been already sequenced in the sequence of length k , $k = 0, 1, \dots$ (assuming $x_{i0} = 0$); the value of the Webster index of model i to be sequenced in position $k + 1$ is $\frac{d_i}{(x_{ik} + 0.5)}$. The local optimization used is the same as the optimization used in the multi-start algorithm.

In this paper we propose to use another index that is evaluated as follows. Let x_{ik} be the number of units of model i that have been already sequenced in the sequence of length k , $k = 0, 1, \dots$ (assuming $x_{i0} = 0$), d_i the number of units of the model i to be sequenced and D the total number of units to be sequenced; the value of our index of the model i to be sequenced in position $k + 1$ is:

$$\frac{(k+1) \cdot d_i}{D} - x_{ik} \quad (1)$$

If there is a tie, then the models with lower d_i are first added in the candidate list.

2.3. Fine-tuning of the algorithm parameters

Fine-tuning the parameters of a metaheuristic algorithm is almost always a difficult task. Although the parameter values are extremely important because the results of the metaheuristic for each problem are very sensitive to them, the selection of parameter values is commonly justified in one of the following ways [15]: 1) “by hand” on the basis of a small number of experiments that are not specifically referenced; 2) by using the general values recommended for a wide range of problems; 3) by using the values reported to be effective in other similar problems; or 4) by choosing values without any explanation.

Adenso-Díaz and Laguna [15] proposed a new technique called CALIBRA for fine-tuning the parameters of heuristic and metaheuristic algorithms. CALIBRA is based on Taguchi’s fractional factorial experimental designs coupled with a local search procedure.

CALIBRA has been chosen for fine-tuning the parameters of our proposed parametric multi-start algorithm, our proposed GRASP algorithm and the GRASP algorithm proposed in [9] (the multi-start algorithm proposed in [9] has not parameters) using a set of 60 representative training instances (generated as explained in Section 3). The following parameter values are obtained: for the parametric multi-start algorithm, $P = 1,500$, and for both GRASP algorithms, size of the candidate list = 3.

The size of the candidate list used in the GRASP algorithm proposed in [9] was 5, but the computational experiment showed that slightly better results are obtained, on average, using the value returned by CALIBRA. Thus, the results shown in the next section are referred only to the ones obtained using a size of the candidate list equal to 3.

3. Computational experiment

Our two proposed algorithms are compared with the PSO algorithm called *DPSOpoi-c_pdyn* [10], which is the most efficient algorithm published to date to solve non-small RTVP instances. We compare also our algorithms with the multi-start and the GRASP algorithms proposed in [9] in order to compare the improvements achieved with the modifications that we have proposed. In what follows in this section, we refer to the multi-start and GRASP algorithms proposed in [9] as *MS-old* and *GR-old*, respectively; and we refer to our proposed multi-start and GRASP algorithms as *MS-new* and *GR-new*, respectively.

The computational experiment was carried out for the same instances used in [9] and [10]. That is, the algorithms ran 740 instances which were grouped into four classes (185 instances in each class) according to their size. The instances in the first class (*CAT1*) were generated using a random value of D (total number of units) uniformly distributed between 25 and 50, and a random value of n (number of models) uniformly distributed between 3 and 15; for the second class (*CAT2*), D was between 50 and 100 and n between 3 and 30; for the third class (*CAT3*), D was between 100 and 200 and n between 3 and 65; and for the fourth class (*CAT4*), D was between 200 and 500 and n between 3 and 150. For all instances and for each model $i = 1, \dots, n$, a random value of d_i (number of units of model i) was between 1 and $\left\lceil \frac{D-n+1}{2.5} \right\rceil$ such that

$\sum_{i=1}^n d_i = D$. All algorithms were coded in Java and the computational experiment was carried out using a 3.4 GHz Pentium IV with 1.5 GB of RAM.

For each instance, all algorithms were run for 50 seconds. Table 1 shows the averages of the RTV values to be minimized for the global of 740 instances and for each class of instances (*CAT1* to *CAT4*).

Table 1. Averages of the RTV values for 50 seconds

| | <i>MS-new</i> | <i>GR-new</i> | <i>DPSOpoi-c_pdyn</i> | <i>MS-old</i> | <i>GR-old</i> |
|---------------|-----------------|-----------------|---------------------------------|------------------|------------------|
| Global | 2,106.01 | 2,308.69 | 4,625.54 | 21,390.40 | 14,168.83 |
| <i>CAT1</i> | 11.56 | 13.00 | 16.42 | 12.08 | 15.47 |
| <i>CAT2</i> | 38.02 | 60.45 | 51.34 | 44.36 | 88.48 |
| <i>CAT3</i> | 154.82 | 270.93 | 610.34 | 226.90 | 510.44 |
| <i>CAT4</i> | 8,219.65 | 8,890.37 | 17,824.04 | 85,278.25 | 56,060.92 |

For the global of all instances, the results of our multi-start and GRASP algorithm are, on average, 54.47% and 50.09%, respectively, better than *DPSOpoi-c_pdyn*, which was to date the best algorithm to solve the RTVP. Moreover, *MS-new* is the best algorithm, on average, for small (*CAT1* and *CAT2*), medium (*CAT3*) and big (*CAT4*) instances. Comparing *MS-new* with *MS-old* by class, we can observe in Table 1 that *MS-new* is 4.30%, 14.29%, 31.77% and 90.36% better than *MS-old* for *CAT1*, *CAT2*, *CAT3* and *CAT4* instances, respectively; and *GR-new* is 15.97%, 31.68%, 46.92% and 84.14% better than *GR-old* for *CAT1*, *CAT2*, *CAT3* and *CAT4* instances, respectively. As we expected, the bigger are the instances, the bigger is the improvement obtained.

To complete the analysis of the results, their dispersion is observed. A measure of the dispersion (let it be called σ) of the RTV values obtained by each metaheuristic mh ($mh = \{MS\text{-}new, GR\text{-}new, DPSOpoi\text{-}c_p\text{-}dyn, MS\text{-}old, GR\text{-}old\}$) for a given instance, ins , is defined as follows:

$$\sigma(mh, ins) = \left(\frac{RTV_{ins}^{(mh)} - RTV_{ins}^{(best)}}{RTV_{ins}^{(best)}} \right)^2 \quad (2)$$

where $RTV_{ins}^{(mh)}$ is the RTV value of the solution obtained with the metaheuristic mh for the instance ins , and $RTV_{ins}^{(best)}$ is, for the instance ins , the best RTV value of the solutions obtained with the four metaheuristics. Table 2 shows the average σ dispersion for the global of 740 instances and for each class of instances.

Table 2. Average σ dispersion regarding the best solution found for 50 seconds

| | <i>MS-new</i> | <i>GR-new</i> | <i>DPSOpoi-c_p-dyn</i> | <i>MS-old</i> | <i>GR-old</i> |
|---------------|---------------|-----------------|----------------------------------|-------------------|-------------------|
| Global | 2.55 | 6,650.83 | 4,931.36 | 202,910.13 | 268,299.58 |
| <i>CAT1</i> | 0.08 | 0.26 | 0.87 | 0.21 | 0.79 |
| <i>CAT2</i> | 0.03 | 1.94 | 0.56 | 0.18 | 6.26 |
| <i>CAT3</i> | 0.05 | 3.07 | 13.80 | 0.50 | 14.18 |
| <i>CAT4</i> | 10.06 | 26,598.04 | 19,710.23 | 811,639.64 | 1,073,177.09 |

Observing the results in Table 2 by class, we can see that *MS-new* has always a very small dispersion far followed by the other algorithms. That means that *MS-new* has a very stable behaviour independently of the size of the instances. For small and medium instances (*CAT1*, *CAT2* and *CAT3*), *GR-new* has also a stable behaviour, but for some big instances (*CAT4*) *GR-new* obtains very bad RTV values. Note that although the RTV values of the *CAT4* instances obtained with *GR-new* are, on average, better than the values obtained with *DPSOpoi-c_p-dyn*, the dispersion of *DPSOpoi-c_p-dyn* is lower than the dispersion of *GR-new*. But comparing the *GR-new* dispersion with the *GR-old* dispersion, we can see *GR-new* has a much more stable behaviour than *GR-old*.

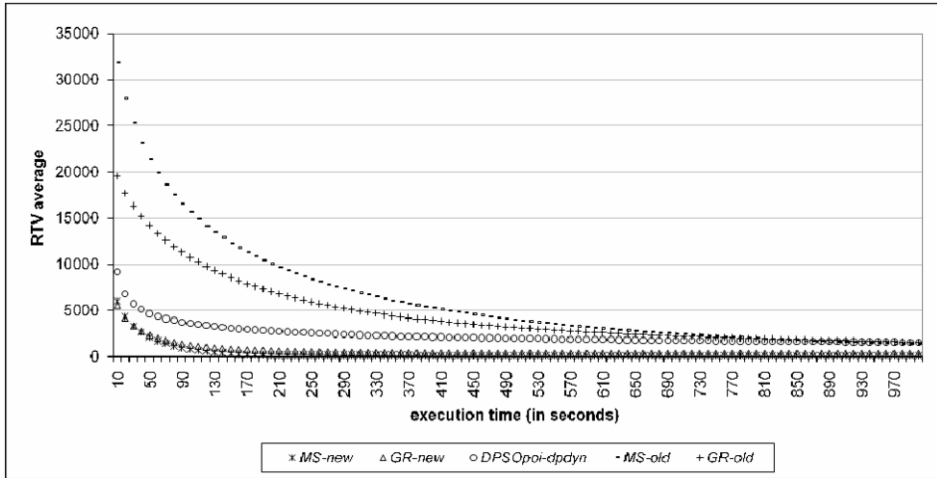
A computing time of 50 seconds may not be long enough to converge for the largest instances (*CAT4* instances). Table 3 shows the averages of the RTV values for the global of all instances and for each class of instances (*CAT1* to *CAT4*) obtained when the algorithms are run for 1000 seconds.

Table 3. Averages of the RTV values for 1,000 seconds

| | <i>MS-new</i> | <i>GR-new</i> | <i>DPSOpoi-c_p-dyn</i> | <i>MS-old</i> | <i>GR-old</i> |
|---------------|---------------|---------------|----------------------------------|-----------------|-----------------|
| Global | 169.25 | 301.90 | 1,537.34 | 1,378.59 | 1,495.12 |
| <i>CAT1</i> | 10.51 | 11.56 | 14.34 | 10.93 | 13.59 |
| <i>CAT2</i> | 31.21 | 50.45 | 46.55 | 35.48 | 75.08 |
| <i>CAT3</i> | 123.27 | 227.50 | 143.96 | 160.67 | 428.86 |
| <i>CAT4</i> | 512.02 | 918.10 | 5,944.51 | 5,307.25 | 5,462.95 |

With 1,000 seconds of execution time, which seems time enough for the convergence of the five algorithms (see Figure 2), *MS-new* is for the global of all instances 43.94%, 88.99%, 87.72% and 88.68% better than the *GR-new*, *DPSOpoi-c_pdyn*, *MS-old* and *GR-old*, respectively; and *GR-new* is 80.36%, 78.10% and 79.81% better than *DPSOpoi-c_pdyn*, *MS-old* and *GR-old*, respectively. Although *DPSOpoi-c_pdyn*, *MS-old* and *GR-old* improve a lot their average results, *MS-new* and *GR-new* are clearly better.

Figure 2. Average of the RTV values obtained over the computing time



Finally, the real-life industrial example presented in [6] was solved using *MS-new*, *GR-new* and *DPSOpoi-c_pdyn*. This example has the following characteristics: $n = 14$, $d = (2, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5)$ and, therefore, $D = 46$. The three algorithms were run ten times with an execution time limit of 1,000 seconds. *MS-new* found the optimal solution in all cases and the minimum, average and maximum computing times were 3.38, 8.74 and 25.03 seconds, respectively. *GR-new* found also the optimal solution in all cases and the minimum, average and maximum computing times were 1.08, 26.53 and 101.86 seconds, respectively. In contrast, *DPSOpoi-c_pdyn* found the optimal solution in only two cases, in computing times of 593.94 and 960.25 seconds.

4. Conclusions and future research

The RTVP occurs in diverse environments as manufacturing, hard real-time systems, operating systems and networks environments. In the RTVP, the aim is to minimize variability in the distances between any two consecutive units of the same model, i.e. to distribute the units as regular as possible. Since it is a NP-hard scheduling optimization problem, heuristic and metaheuristic methods are needed.

This paper is an extension of the work started in [9], in which one multi-start, one GRASP and four PSO algorithms were proposed. New PSO algorithms to solve the RTVP have been published in [10]. The best of them, *DPSOpoi-c_pdyn*, obtains the best

results to date. In this paper, an improved multi-start algorithm, *MS-new*, and an improved GRASP algorithm, *GR-new*, are proposed to solve the RTVP.

The computational experiment shows clearly that the proposed algorithms obtain, on average, strongly better solutions than *DPSOpoi-c_pdyn* independently of the size of the RTVP instance. Moreover, *MS-new*, the proposed algorithm that obtains the best solutions, has always a very stable behaviour. Instead, *GR-new* and *DPSOpoi-c_pdyn* have not. Therefore, it is advisable to use always *MS-new* for solving the RTVP.

Although the RTVP is hard to solve, it is interesting to try to solve it by means of exact procedures to know the largest size of the RTVP instances that can be solved optimally in a practical computing time. The two exact procedures proposed in the literature are MILP models [1, 8]. Since the use of Constraint Programming (CP) to solving the RTVP has not been proposed yet in the literature, applying CP to RTVP seems a promising future line of research.

ACKNOWLEDGEMENTS

The authors wish to express their gratitude to two anonymous reviewers for their valuable comments, which have improved the quality of this paper.

References

- [1] Corominas, A., Kubiak, W. and Moreno, N. (2007) 'Response time variability', *Journal of Scheduling*, Vol. 10, pp. 97-110.
- [2] Monden, Y. (1983) 'Toyota Production Systems', *Industrial Engineering and Management Press*, Norcross, GA.
- [3] Waldspurger, C.A. and Weihl, W.E. (1995) 'Stride Scheduling: Deterministic Proportional-Share Resource Management', *Technical Report MIT/LCS/TM-528*, Massachusetts Institute of Technology, MIT Laboratory for Computer Science.
- [4] Dong, L., Melhem, R. and Mosse, D. (1998) 'Time slot allocation for real-time messages with negotiable distance constraints requirements', *Fourth IEEE Real-Time Technology and Applications Symposium (RTAS'98)*, Denver, CO. pp.131-136.
- [5] Anily, S., Glass, C.A. and Hassin, R. (1998) 'The scheduling of maintenance service', *Discrete Applied Mathematics*, Vol. 82, pp.27-42.
- [6] Herrmann, J.W. (2007) 'Generating Cyclic Fair Sequences using Aggregation and Stride Scheduling', *Technical Report*, University of Maryland, USA.
- [7] Miltenburg, J. (1989) 'Level schedules for mixed-model assembly lines in just-in-time production systems', *Management Science*, Vol. 35, No. 2, pp. 192-207.
- [8] Corominas, A., Kubiak, W. and Pastor, R. (2006) 'Solving the Response Time Variability Problem (RTVP) by means of mathematical programming', *Working paper IOC-DT*, Universitat Politècnica de Catalunya, Spain.
- [9] García, A., Pastor, R. and Corominas, A. (2006) 'Solving the Response Time Variability Problem by means of metaheuristics', *Special Issue of Frontiers in Artificial Intelligence and Applications on Artificial Intelligence Research and Development*, Vol. 146, pp.187-194.
- [10] García-Villoria, A. and Pastor, R. (2007) 'Introducing dynamic diversity into a discrete particle swarm optimization', *Computers & Operations Research*, In Press, Corrected Proof, Available online 7 December 2007, doi:10.1016/j.cor.2007.12.001.
- [11] Boender, C.G.E., Rinnooy, A.H.G., Stougie, L. and Timmer, G.T. (1982) 'A Stochastic Method for Global Optimization', *Mathematical Programming*, Vol. 22, pp.125-140.
- [12] Marti, R. (2003), 'Multi-start methods', *Handbook of Metaheuristics*, Glover and Kochenberger (eds.), Kluwer Academic Publishers, pp.355-368.
- [13] Hoos, H. and Stützle, T. (2005) *Stochastic local search: foundations and applications*, Morgan Kaufmann Publishers, San Francisco.

- [14] Feo, T.A. and Resende, M.G.C. (1989) 'A probabilistic heuristic for a computationally difficult set covering problem', *Operations Research Letters*, Vol. 8, pp.67-81.
- [15] Adenso-Díaz, B. and Laguna, M. (2006) 'Fine-tuning of algorithms using fractional experimental designs and local search', *Operations Research*, Vol. 54, pp. 99-114.

An algorithm based on structural analysis for model-based fault diagnosis

Esteban R. GELSO, Sandra M. CASTILLO , and Joaquim ARMENGOL

IIIA - Universitat de Girona, Campus de Montilivi, Girona, E-17071, Spain

{esteban.gelso, sandra.castillo, joaquim.armengol}@udg.edu

Abstract. This paper presents a diagnosis system in which, an algorithm automatically finds all minimal structurally overdetermined (MSO) sets in a structural model of a system. It combines a first set of MSO sets to get the additional MSO sets, which were obtained after determining a complete matching between equations and unknown variables. Finding the complete set is useful for the diagnosis task increasing the fault isolability due to the fact that it can provide different signatures to each fault. The efficiency of the algorithm for finding all MSO sets is illustrated using the DAMADICS benchmark, which is a pneumatically-actuated control valve.

Keywords. Model-based fault diagnosis, structural analysis, fault isolation, redundancy, complete matching

Introduction

When faults occur, consequences can be catastrophic in terms of economic costs, human losses and environmental impact, for instance. Fault detection and diagnosis for complex plants can minimize downtime, increase productivity and the safety of plant operations, reduce manufacturing costs and assure product quality. They can be approached from different perspectives according to the nature of the knowledge available about the system and faults that occur in the system such as knowledge-based reasoning, case-based reasoning, machine learning, and model-based approaches [1,2]. The so-called model-based diagnosis (MBD) approaches rest on the use of an explicit model of the system to be diagnosed [3,4,5]. The occurrence of a fault is captured by discrepancies (residuals) between the observed behavior and the behavior that is predicted by the model. Thus, model-based fault detection consists of the detection of faults in the processes, actuators, and sensors by exploiting the dependencies between different measurable signals. These dependencies are expressed by mathematical process models.

As much the complexity of the system increases, the greater is the difficulty of designing efficient diagnosis procedures. Therefore, an important task to develop is the design of detection and isolation tests in order to overcome faults quickly by obtaining a fast and correct diagnostic. What it really means is to obtain the complete set of overconstrained, or overdetermined, subsystems, and thus, to achieve an optimal diagnostic.

These fault tests can take the form of parity relations or analytical redundancy relations (ARR) [6], and can be deduced by means of Structural Analysis (Section 1). Once they are designed (off-line), the fault detection procedure checks on-line the consistency

of the observations with respect to every of these tests. When discrepancies occur between the modeled behavior and the observations (non-zero residual signals), the fault isolation procedure identifies the system component(s) which is (are) suspected of causing the fault [5].

Being the obtainment of minimal structurally overdetermined (MSO) sets [7] a really complex task, some authors have developed other approaches to deal with this issue. A Matlab[®] toolbox called SaTool is presented in [8], in which the overdetermined sets are obtained by means of the constraints that are not involved in the complete matching. This toolbox, which does not deliver all the MSO sets, is the starting point of the algorithm proposed in this paper. In [7], an algorithm based on a top-down approach is presented. It starts with the entire model and then reduces the size of it step by step until an MSO set remains. Moreover, in e.g. [9,10,11,12], information about causality can be taken into account to discard unachievable overdetermined sets, by a successive elimination of the unknown variables in [10,11].

This paper presents an approach for automatically finding all MSO sets in a structural model of a system. It combines a first set of MSO sets to get the additional MSO sets, which were obtained after determining a complete matching between equations and unknown variables. An in-depth description of the proposed algorithm features are presented in Section 2.

Finally, the efficiency of the algorithm for finding all MSO sets was demonstrated by applying it to a model of a pneumatically-actuated control valve (See Section 3).

1. STRUCTURAL ANALYSIS

In the structural analysis [13], the structure of a model M (or the structural model) of the system $(M, X \cup Z)$ is represented by a bipartite graph $G(M, X \cup Z)$ (or equivalently its biadjacency matrix) with variables and constraints as node sets. X is the set of unknown variables and Z is the set of known variables. There is an edge $(c_i, v_j) \in \mathcal{E}$ (set of edges) connecting a constraint $c_i \in M$ and an unknown variable $v_j \in X \cup Z$ if v_j is included in c_i [6].

Considering the following definitions:

Definition 1 *Matching.* A matching \mathcal{M} is a subset of edges \mathcal{E} such that with $e_1 = (c_{i1}, v_{j1})$, $e_2 = (c_{i2}, v_{j2})$, $\forall e_1, e_2 \in \mathcal{M} : e_1 \neq e_2 \Rightarrow c_{i1} \neq c_{i2} \wedge v_{j1} \neq v_{j2}$.

Definition 2 *Complete matching on X .* A matching is complete on X if $|\mathcal{M}| = |X|$.

the central idea in the structure graph approach is to match all unknown variables using available constraints and known variables, if possible. If successful, the matching will identify over-determined subgraphs (those which contain more equations than unknown variables) that can be used to obtain analytical redundancy relations in the system. In particular, the minimal over-determined subgraphs are useful for this task (a structurally overdetermined set is minimal if no proper subset is a structurally overdetermined set [7]). Several algorithms exist to find complete matchings on unknown variables in a structure graph. In this work, the ranking algorithm (Algorithm 1) is used to achieve a complete matching on the unknown variables [6]. The input of this algorithm is the biadjacency matrix S of $G(M, X \cup Z)$. In this matrix each row represents a constraint and each column a variable. $S(i, j) = 1$ means that variable x_j appears in constraint c_i .

Algorithm 1 Ranking Algorithm**Require:** Biadjacency matrix

- 1: Mark all known variables, $i=0$;
- 2: Find all constraints with exactly one unmarked variable.
Associate rank i with these constraints.
Mark these constraints and the associated variable;
- 3: If there are unmarked constraints whose variables are all marked, mark them and connect them with the pseudovvariable zero;
- 4: set $i = i+1$;
- 5: if there are unmarked variables or constraints, continue with step 2;
- 6: **return** Ranking of the constraints

In the first step of Algorithm 1, all known variables are marked and all unknown variables remain unmarked. Then every constraint that contains at most one unmarked variable is assigned rank 0. The constraint is matched for the unmarked variables, and the variable is marked. This step is repeated with an increasing rank number, until no new variables can be matched. Every matched variable has associated a number, or rank, which can be interpreted as the number of steps needed to calculate an unknown variable from the known ones.

Unmatched constraints are used to obtain a first family of MSO sets, which will be called Basic MSO sets. Each MSO set is obtained from an unmatched constraint c_i by backtracking unknown variables of c_i through constraints to which they were matched [6].

2. ALGORITHM FOR FINDING ALL MSO SETS

In this section, a new algorithm will be presented for finding all MSO sets. It is based on the fact that the basic MSO sets can be combined in order to get more MSO sets, using the information from one complete matching instead of finding all complete matchings.

Algorithm 2 is based on combining the basic MSO sets (the collection of basic MSO sets is called \mathcal{C}_{MSO_1} in the algorithm). A structurally overdetermined set can be obtained from the elimination of at least one shared equation from the set of equations of two MSO sets, and this operation is called a combination. The combinations of MSO sets can be minimal or not. This algorithm finds only the minimal ones.

Algorithm 2 Algorithm to find all MSO sets**Require:** Complete matching

- 1: $\mathcal{C}_{MSO_1} \leftarrow$ Basic MSO sets;
- 2: $i = 1$;
- 3: **while** $i <$ number of MSO sets in \mathcal{C}_{MSO_1} , or, \mathcal{C}_{MSO_i} is not empty **do**
- 4: $\mathcal{C}_{MSO_{i+1}} := \text{Combine}(\mathcal{C}_{MSO_i}, \mathcal{C}_{MSO_1})$;
- 5: set $i=i+1$;
- 6: **end while**
- 7: $\mathcal{C}_{MSO} := (\mathcal{C}_{MSO_1} \dots \mathcal{C}_{MSO_i})$;
- 8: **return** Complete set of MSO sets: \mathcal{C}_{MSO}

Function ‘Combine’, which is presented as Algorithm 3, leads to obtain the new MSO sets after combining two collections of MSO sets. Step four in the algorithm is very important. It can be tackled in a brute-force way, which can result in a combinatorial explosion. This method avoids this by using the rank information provided by Algorithm 1 applied to the biadjacency matrix of $G(M_{SD}, X \cup Z)$. Then it removes one shared constraint at a time, and the corresponding matched constraints used only to calculate the unknown variables of the shared constraint.

Algorithm 3 Combine function

```

1:  $\mathcal{C}_{MSO_{ab}} = \text{Combine}(\mathcal{C}_{MSO_a}, \mathcal{C}_{MSO_b})$ 
2: for all set  $MSO_a$  in  $\mathcal{C}_{MSO_a}$  do
3:   for all set  $MSO_b$  in  $\mathcal{C}_{MSO_b}$  do
4:     if shared constraints set of  $MSO_a$  and  $MSO_b$  is not void, and,  $MSO_a$  and  $MSO_b$  do not share the same unmatched constraint then
5:       Remove one or more shared constraints; Check if the new structurally overdetermined set is minimal; If it is minimal, add to  $\mathcal{C}_{MSO_{ab}}$ ;
6:     end if
7:   end for
8: end for
9: return  $\mathcal{C}_{MSO_{ab}}$ 
  
```

As an example, consider the following biadjacency matrix:

| | x_1 | x_2 | x_3 | x_4 | x_5 | x_6 | x_7 |
|-------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| c_1 | 1^\diamond | | | | | | |
| c_2 | 1 | 1^\diamond | | | | | |
| c_3 | | 1 | 1^\diamond | | | | |
| c_4 | | | | 1^\diamond | 1 | | |
| c_5 | | | | | 1^\diamond | | |
| c_6 | | | | | | 1^\diamond | 1 |
| c_7 | | | | | | | 1^\diamond |
| c_8 | | | 1 | 1 | | | |
| c_9 | | | 1 | | | 1 | |

Matched constraints are c_1 to c_7 . Therefore two basic MSO sets can be found using c_8 and c_9 . The first is $\{c_1, c_2, c_3, c_4, c_5, c_8\}$, and the second is $\{c_1, c_2, c_3, c_6, c_7, c_9\}$. The shared constraints in both are $\{c_1, c_2, c_3\}$, and then, seven different possibilities of removing are possible. Instead of this, using the information of the matched variables, we can see that c_3 (rank 2) was matched using c_2 (rank 1), and c_2 using c_1 (rank 0). Removing c_3 , in this example also implies that c_1 and c_2 can be removed because they are not used for another calculation. So the third MSO set is $\{c_4, c_5, c_6, c_7, c_8, c_9\}$.

2.1. Computational complexity

The computational complexity can be studied in two parts:

(i) for the matching algorithm used to find the basic MSO sets. The ranking algorithm has complexity $O(nm)$ where n is the number of constraints and m is the number of unknown variables [8].

(ii) for the algorithm used to combine the basic MSO sets. Being φ the structural redundancy of a model, the φ basic MSO sets are combined, using Algorithm 2, in groups

of 2 to φ (in the worst case). For each combination, there are at most r shared constraints to be removed. r is less than or equal to m . For a worst case, the function of operations can be expressed by the following function

$$\sum_{k=2}^{\varphi} \binom{\varphi}{k} r^{k-1} \leq \sum_{k=2}^{\varphi} \binom{\varphi}{k} m^{k-1} < \sum_{k=2}^{\varphi} \binom{\varphi}{k} m^{\varphi-1} = m^{\varphi-1} (2^{\varphi} - \varphi - 1) \quad (1)$$

Hence, for a fixed order of structural redundancy, the computational complexity of the algorithm is polynomial in the number of unknowns. This condition makes this algorithm suitable for real-world systems with a large number of unknown variables and constraints, but low structural redundancy (which depends on the number of available sensors) [7].

2.2. Fault isolability analysis

Using the information of the faults which influence each constraint, the (theoretical) fault signature matrix can be derived. The fault signature of each possible residual, represents the set of faults which affects at least one of the constraints used to generate this residual.

Definition 3 *The Fault signature matrix is a binary table in which each line corresponds to a residual and each column to a fault. A 0 in the position (i, j) indicates that the occurrence of the fault f_j does not affect the residual i , and a 1 otherwise [5].*

As defined in [14], in order to easily visualize the isolability property of faults, the isolability matrix is computed. Two faults are structurally isolable if their signatures are different.

Definition 4 *The isolability matrix is a square matrix where each row and each column correspond to a fault. A 1 in the position (i, j) indicates that fault i is not isolable from fault j .*

3. APPLICATION EXAMPLE

The application example to test the proposed approach, deals with an industrial smart actuator, proposed as an FDI benchmark in the European DAMADICS project. As described in [15], the valve consists of three main components: (i) control and by-pass valves, (ii) a spring-and-diaphragm pneumatic servo-motor, and (iii) a positioner. A general scheme of the valve is shown in Fig. 1.

Measured variables, indicated by circles in the figure, are the valve plug position X' , the fluid flow F' , fluid temperature T'_1 , upstream and downstream pressure of the valve P'_1, P'_2 , the transducer chamber pressure P'_s and the positioner supply pressure P'_z .

The set of 19 faults acting on the valve and its components is described in [16]. In this paper the set of faults was extended by 4 additional faults to consider all measurement faults.

The algorithm presented in Section 2 leads to 29 MSO sets, summarized in Table 2, after combining 4 basic MSO sets (see Table 3).

Table 2. MSO sets of the application example.

| n° MSO sets | n° MSO sets |
|--|--|
| 1 $\{c_1, c_2, c_8, c_{10}, c_{11}, c_{12}, c_{14}\}$ | 16 $\{c_3, c_4, c_6, c_7, c_9, c_{10}, c_{11}, c_{12}, c_{13}, c_{14}, c_{15}\}$ |
| 2 $\{c_4, c_8, c_{13}, c_{14}, c_{15}\}$ | 17 $\{c_1, c_2, c_3, c_4, c_6, c_7, c_8, c_9, c_{11}, c_{12}, c_{13}, c_{15}\}$ |
| 3 $\{c_5, c_{15}\}$ | 18 $\{c_1, c_2, c_3, c_4, c_6, c_7, c_8, c_9, c_{10}, c_{12}, c_{13}, c_{15}\}$ |
| 4 $\{c_3, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{12}\}$ | 19 $\{c_1, c_2, c_3, c_4, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{13}, c_{15}\}$ |
| 5 $\{c_1, c_2, c_4, c_{10}, c_{11}, c_{12}, c_{13}, c_{14}, c_{15}\}$ | 20 $\{c_1, c_2, c_3, c_4, c_6, c_7, c_9, c_{11}, c_{12}, c_{13}, c_{14}, c_{15}\}$ |
| 6 $\{c_1, c_2, c_4, c_8, c_{10}, c_{11}, c_{12}, c_{13}, c_{15}\}$ | 21 $\{c_1, c_2, c_3, c_4, c_6, c_7, c_9, c_{10}, c_{12}, c_{13}, c_{14}, c_{15}\}$ |
| 7 $\{c_1, c_2, c_3, c_6, c_7, c_9, c_{10}, c_{11}, c_{12}, c_{14}\}$ | 22 $\{c_1, c_2, c_3, c_4, c_6, c_7, c_9, c_{10}, c_{11}, c_{13}, c_{14}, c_{15}\}$ |
| 8 $\{c_1, c_2, c_3, c_6, c_7, c_8, c_9, c_{11}, c_{12}, c_{14}\}$ | 23 $\{c_1, c_2, c_3, c_4, c_6, c_7, c_9, c_{10}, c_{11}, c_{12}, c_{13}, c_{15}\}$ |
| 9 $\{c_1, c_2, c_3, c_6, c_7, c_8, c_9, c_{10}, c_{12}, c_{14}\}$ | 24 $\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{11}, c_{12}, c_{13}\}$ |
| 10 $\{c_1, c_2, c_3, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{14}\}$ | 25 $\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_9, c_{11}, c_{12}, c_{13}, c_{14}\}$ |
| 11 $\{c_4, c_5, c_8, c_{13}, c_{14}\}$ | 26 $\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_9, c_{11}, c_{12}, c_{13}, c_{14}\}$ |
| 12 $\{c_1, c_2, c_4, c_5, c_{10}, c_{11}, c_{12}, c_{13}, c_{14}\}$ | 27 $\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_9, c_{10}, c_{12}, c_{13}, c_{14}\}$ |
| 13 $\{c_1, c_2, c_4, c_5, c_8, c_{10}, c_{11}, c_{12}, c_{13}\}$ | 28 $\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_9, c_{10}, c_{11}, c_{13}, c_{14}\}$ |
| 14 $\{c_3, c_4, c_5, c_6, c_7, c_9, c_{10}, c_{11}, c_{12}, c_{13}, c_{14}\}$ | 29 $\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_9, c_{10}, c_{11}, c_{12}, c_{13}\}$ |
| 15 $\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_8, c_9, c_{10}, c_{11}, c_{13}, c_{14}\}$ | |

Table 3. Results of Algorithm 2 applied to the application example

| Collection | Number of MSO sets |
|-----------------------|--------------------|
| \mathcal{C}_{MSO_1} | 4 |
| \mathcal{C}_{MSO_2} | 8 |
| \mathcal{C}_{MSO_3} | 10 |
| \mathcal{C}_{MSO_4} | 7 |
| \mathcal{C}_{MSO} | 29 |

The fault isolability analysis matrix for the 4 basic MSO sets is given as shown in Table 4. From this table, it is seen that the first five blocks of 4, 5, 5, 3 and 2 faults, respectively, show faults that are not isolable from the other faults in the group, but isolable from other faults in different groups. f_{13} , f_{15} and f_{23} are isolable from all other faults.

When considering the complete collection of MSO sets, a fault isolability matrix as in Table 5 is obtained. As it can be seen, now it is possible to isolate f_{10} , f_{14} , f_{20} , f_{21} and f_{22} from each other. Moreover, f_2 is not isolable from f_3 and f_3 is not isolable from f_2 .

| | f_{11} | f_8 | f_4 | f_1 | f_{22} | f_{21} | f_{20} | f_3 | f_2 | f_{19} | f_{18} | f_7 | f_6 | f_5 | f_{16} | f_{12} | f_9 | f_{14} | f_{10} | f_{13} | f_{15} | f_{23} |
|----------|----------|-------|-------|-------|----------|----------|----------|-------|-------|----------|----------|-------|-------|-------|----------|----------|-------|----------|----------|----------|----------|----------|
| f_{11} | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_8 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_4 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{22} | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{21} | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{20} | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_3 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_2 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{19} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{18} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{16} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| f_{12} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| f_9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| f_{14} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| f_{10} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| f_{13} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| f_{15} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| f_{23} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Table 4. Fault isolability analysis matrix.

| | f_{11} | f_8 | f_4 | f_1 | f_{22} | f_{21} | f_{20} | f_3 | f_2 | f_{19} | f_{18} | f_7 | f_6 | f_5 | f_{16} | f_{12} | f_9 | f_{14} | f_{10} | f_{13} | f_{15} | f_{23} |
|----------|----------|-------|-------|-------|----------|----------|----------|-------|-------|----------|----------|-------|-------|-------|----------|----------|-------|----------|----------|----------|----------|----------|
| f_{11} | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_8 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_4 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{22} | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{21} | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{20} | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{19} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{18} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| f_{16} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| f_{12} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| f_9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| f_{14} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| f_{10} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| f_{13} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| f_{15} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| f_{23} | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

Table 5. Fault isolability analysis matrix using the 29 MSO sets.

4. CONCLUSIONS

The main contribution of this paper is the proposed algorithm to obtain the whole set of MSO sets of the system. When talking about over constrained systems, the task of obtaining the complete set of minimal mathematical relations (to retrieve an optimal diagnostic) becomes a complex task.

As it is shown, the proposed algorithm became one of the alternatives to perform the task of design fault detection and isolation test, by founding all the subsystems could be diagnosed in the system. This algorithm is based on the ranking algorithm and it completes the previous approach explained in [8], which does not deliver all the MSO sets.

Features of this algorithm were presented and its efficiency was demonstrated by the application example. The computational complexity of the proposed algorithm is polynomial in the number of unknowns making this condition suitable for real-world systems with a large number of unknown variables and constraints.

Finally, in future we intend to compare the algorithm to find all MSO sets with other solutions cited in this article, in terms of e.g. computational complexity and real world performance.

5. Acknowledgements

This work has been funded by the Spanish Government (Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica, Ministerio de Educación y Ciencia) through the coordinated research project grant No. DPI2006-15476-C02-02, by the grant No. 2005SGR00296 and the Departament d'Innovació, Universitats, i Empresa of the Government of Catalonia.

References

- [1] K. Balakrishnan and V. Honavar, "Intelligent diagnosis systems," *Journal of Intelligent Systems*, vol. 8, no. 3, pp. 239–290, 1998.
- [2] V. Venkatasubramanian, R. Rengaswamy, K. Yin, and S. Kavuri, "A review of process fault detection and diagnosis: Part I: Quantitative model-based methods," *Computers and Chemical Engineering*, vol. 27, pp. 293–311, 2003.
- [3] J. J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*. Marcel Dekker, 1998.
- [4] W. Hamscher, L. Console, and J. de Kleer, *Readings in Model-Based Diagnosis*. San Francisco, CA, USA: Morgan Kaufman, 1992.
- [5] M. O. Cordier, P. Dague, F. Levy, J. Montmain, M. Staroswiecki, and L. Trave-Massuyes, "Conflicts versus analytical redundancy relations: a comparative analysis of the model based diagnosis approach from the artificial intelligence and automatic control perspectives," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, pp. 2163–2177, October 2004.
- [6] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault-Tolerant Control*. Springer, 2nd ed., 2006.
- [7] M. Krysander, J. Åslund, and M. Nyberg, "An efficient algorithm for finding minimal over-constrained sub-systems for model-based diagnosis," *IEEE Trans. on Systems, Man, and Cybernetics – Part A: Systems and Humans*, vol. 38, no. 1, 2008.
- [8] M. Blanke and T. Lorentzen, "Satoool - a software tool for structural analysis of complex automation systems," in *6th IFAC Symposium on Fault Detection, Supervision and Safety of Technical Processes*, (Beijing), 2006.

- [9] B. Pulido and C. Alonso González, "Possible conflicts: A compilation technique for consistency-based diagnosis," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, no. 5, pp. 2192–2206, 2004.
- [10] L. Travé-Massuyès, T. Escobet, and X. Olive, "Diagnosability analysis based on component supported analytical redundancy relations," *IEEE Trans. on Systems, Man and Cybernetics, Part A : Systems and Humans*, vol. 36, no. 6, 2006.
- [11] S. Ploix, M. Désinde, and S. Touaf, "Automatic design of detection tests in complex dynamic systems," in *16th IFAC World Congress*, (Prague), 2005.
- [12] D. Dustegor, V. Cocquempot, and M. Staroswiecki, "Structural analysis for residual generation: towards implementation," *Control Applications, 2004. Proceedings of the 2004 IEEE International Conference on*, vol. 2, pp. 1217–1222, 2004.
- [13] M. Staroswiecki, *Structural analysis for fault detection and isolation and for fault tolerant control*, ch. Encyclopedia of Life Support Systems. Fault Diagnosis and Fault Tolerant Control, 2002.
- [14] D. Düşteğör, E. Frisk, V. Cocquempot, M. Krysander, and M. Staroswiecki, "Structural analysis of fault isolability in the DAMADICS benchmark," *Control Engineering Practice*, vol. 14, no. 6, pp. 597–608, 2006.
- [15] M. Bartyś, P. R., S. M., de las Heras S., and Q. J., "Introduction to the damadics actuator fdi benchmark study," *Control Engineering Practice*, vol. 14, no. 6, pp. 577–596, 2006.
- [16] M. Bartyś and S. M., "Using damadics actuator benchmark library (dablib)," *Final*, v.1.22, 2002.

This page intentionally left blank

Knowledge and Information Systems

This page intentionally left blank

Knowledge Discovery with Explained Case-Based Reasoning

Eva ARMENGOL

*IIIA, Artificial Intelligence Research Institute
CSIC, Spanish Council for Scientific Research
Campus UAB, 08193 Bellaterra, Barcelona (Spain)
e-mail: eva@iiia.csic.es*

Abstract.

The goal of Knowledge Discovery is to extract knowledge from a set of data. Most common techniques used in knowledge discovery are clustering methods, whose goal is to analyze a set of objects and obtain clusters based on the similarity among these objects. A desirable characteristic of clustering results is that these should be easily understandable by domain experts. In fact, these are characteristics that exhibit the results of eager learning methods (such as ID3) and lazy learning methods when used for building lazy domain theories. In this paper we propose LazyCL, a procedure using a lazy learning method to produce explanations on clusters of unlabeled cases. The analysis of the relations among these explanations converges to a correct clustering of the data set.

Keywords. Knowledge Discovery, Clustering, CBR, Explanations.

1. Introduction

Case-Based Reasoning (CBR) methods predict the classification of a problem based on its similarity to already solved cases. One of the key points of CBR systems is the measure used to assess the similarity among cases, since the final classification of a new problem depends on it. However this similarity measure often is not easy to understand and as a consequence, experts (that commonly are not mathematicians) could not be totally convinced about the results produced by the system. For this reason, in recent years has been an increasing interest on approaches addressed to explain CBR results in a satisfactory way (see [18]). One of these approaches is LID, a lazy learning method we introduced in [3]. During the problem solving process LID builds an explanation justifying the classification of a new problem. This explanation is in fact, a generalization of relevant features shared by both problem and cases. On the other hand, in [1] we argued that generalizations can be seen as explanations since that they commonly contain problem features useful for classifying problems. This is the case of prototypes from PRO-TOS [4], Generalized Cases [5], and Lazy Decision Trees [10]. Also, *Explanation-Based Learning (EBL)* methods [17] generalize a particular example to obtain a domain rule that can be used for solving unseen problems. Our point is that explanations produced

by lazy learning methods, for instance LID, should be considered as domain rules in the same way as generalizations are. Thus, the set of explanations could be considered as a *lazy* domain theory.

Commonly, domain theories are built using eager learning methods (such as ID3 [19]). Eager learning methods build discriminant descriptions for classes, therefore the union of these discriminant descriptions cover all the space of known examples. Instead, lazy domain theories cover only zones around each new problem, therefore this could result in “holes” when describing the domain. In [2] we compared lazy domain theories formed by sets of explanations from LID with the eager theory build by the ID3 method [19]. In our experiments we showed that for some domains both eager and lazy theories have a similar predictivity. The difference is that because of the explanations composing the lazy domain theory are more specific than eager rules, there is a high percentage of unseen problems that the lazy theory cannot classify, although the classification, when it is proposed, is most of times correct.

In the current paper we propose to exploit the concept of lazy domain theory for knowledge discovery. Although lazy domain theories are formed by local rules, this information is very valuable to the experts to obtain a picture of some parts of the domain. Frawley et al [9] defined *Knowledge Discovery* as “the non-trivial extraction of implicit, unknown and potentially useful information from data”. In fact, we want to support domain experts in building a domain theory, producing explanations (generalizations) that could be easily understood and giving them the opportunity to systematically analyze the proposed classes. However knowledge discovery problems cannot be directly solved by means of either lazy or eager learning methods since most of them need to know in advance the class label of the domain objects. Most common techniques used in knowledge discovery are *clustering* methods whose goal is to analyze a set of objects and obtain clusters based on the similarity among objects. On the other hand, lazy learning methods cannot be used for clustering because cases are not labeled. Therefore, how a lazy learning method could be used for knowledge discovery? Our proposal is to randomly cluster the domain objects and considering these clusters as solution classes. On these classes a lazy learning method can be used in order to obtain explanations that can be seen as domain rules. We call this procedure **LazyCL** and we used it on some data sets from the UCI Repository [7] to show that the analysis of relations among explanations is a good basis for knowledge discovery.

The paper is structured as follows. Next section describes the **LazyCL** procedure. Section 3 describes the experiments we carried out and analyzes some particular results on UCI data sets. Section 4 comments some works related with our approach.

2. Using Explanations for Knowledge Discovery

Let us suppose the following scenario. Domain experts have available a set of object descriptions and they would extract classes from them. These classes should be reasonable from their point of view, therefore it is necessary to give an explanation of the clustering. Moreover, experts would obtain some kind of domain theory allowing them to univocally characterize the classes, in the same sense that supradiscriminant descriptions from eager learning method do. In this scenario, the use of a supervised learning method is not possible because the actual classes of objects are not known. Therefore, we propose **LazyCL** producing discriminant descriptions of clusters according to the following 4 steps:

1. *Creation of clusters.* Clusters are randomly created to obtain a case base where each case c_i has associated a solution class Cl_k . In this step, both the cardinality of clusters and the number of clusters are randomly decided.
2. *Obtention of explanations.* In this second step, the LID method is used with leave-one-out to obtain explanations. Each case c_i is classified by LID as belonging to a cluster Cl_j . In addition, LID gives an explanation d_j of such classification. A possible situation is that the method cannot univocally classify a case into a cluster, in other words, the method gives the description (explanation) d_{ji} that is satisfied by cases of two or more clusters. In such situation, a new cluster Cl_{ij} is created and the description d_{ij} will be associated with it. The result of this second step is that each cluster Cl_i (either from the original clustering or from the newly created during the problem solving process) has associated one or more explanations $d_{i1}...d_{in}$ describing it.
3. *Redundancy elimination* step to detect relationships among descriptions. Let us suppose that the description d_{ji} is more general than description d_{jk} , in such situation the description d_{jk} (the most specific of both) is rejected. For instance, let us suppose that a cluster Cl_i is described by the following two descriptions (from the Glass data set of the UCI repository [7]):

- $d_1 : (Ri \in [1.52, 1.73]) \text{ and } (Na \leq 14.065) \text{ and } (Al \leq 1.39) \text{ and } (K \in [0.055, 0.615]) \text{ and } (Ca \in [8.315, 10.075])$
- $d_2 : (Ri \in [1.52, 1.73]) \text{ and } (K \in [0.055, 0.615]) \text{ and } (Ca \in [8.315, 10.075])$

During this step d_1 will be eliminated since it is more specific than d_2 and, therefore, all the cases satisfying d_1 will also satisfy d_2 . Notice that at the end of this step, each cluster Cl_j is described by a set of discriminant descriptions $d_{j1}...d_{jn}$ where each d_{ji} describes a subset of the cases included in the cluster.

4. *Merging of clusters.* The process is the same as that performed during the previous step, but now is applied to the descriptions of different clusters. In other words, the goal of this step is to detect relationships among the descriptions of the different clusters.

In step 2, we have used the LID method as problem solver. *Lazy Induction of Descriptions (LID)* is a lazy learning method for classification tasks. LID determines which are the most relevant attributes of a problem and searches in a case base for cases sharing these relevant attributes. The problem p is classified when LID finds a set of relevant attributes shared by a subset of cases all belonging to the same solution class C_i . Then LID classifies the problem as belonging to C_i . We call *similitude term* the description formed by these relevant attributes and *discriminatory set* the set of cases satisfying the similitude term. In fact, a similitude term is a generalization of both p and the cases in the discriminatory set. Figure 1 shows the LID algorithm (see [3] for details).

LID has two possible stopping situations: (1) all the cases in the discriminatory set belong to the same solution class, and (2) there is no attribute allowing to specialize the similitude term. When LID finishes with an explanation that only is satisfied by cases belonging to a class, the explanation can be interpreted as a partial description of the class in the same sense than eager methods. Nevertheless, when LID finishes because the current similitude term cannot be further specialized, it contains the important attributes


```

Function LID ( $p, S_D, D, C$ )
   $S_D :=$  Discriminatory-set ( $D$ )
  if stopping-condition( $S_D$ )
    then return class( $S_D$ )
  else  $f_d :=$  Select-attribute ( $p, S_D, C$ )
        $D' :=$  Add-attribute( $f_d, D$ )
        $S_{D'} :=$  Discriminatory-set ( $D', S_D$ )
       LID ( $S_{D'}, p, D', C$ )
  end-if
end-function

```

Figure 1. The LID algorithm. p is the problem to be solved, D is the similitude term, S_D is the discriminatory set associated to D , C is the set of solution classes.

used to classify a problem. In such situation the similitude term can be interpreted as a justification or *explanation* of why the problem has been classified in C_i .

Concerning step 4 of LazyCL, let C_i and C_j be two clusters, their respective descriptions can be related in the following alternative ways:

- Let us suppose that Cl_i and Cl_j are described descriptions d_i and d_j , respectively. If d_i is more general than d_j both Cl_j and d_j are rejected.
- Let us suppose that Cl_i is described by one description, namely d_i , and Cl_j is described by several descriptions, namely $d_{j1}...d_{jn}$. If one of these descriptions, say d_{jk} , is more general than d_i then both Cl_i and d_i are rejected.
- Let us suppose that Cl_i is described by one description, namely d_i , and Cl_j is described by several descriptions, namely $d_{j1}...d_{jn}$. If d_i is more general than d_{jk} then d_{jk} is eliminated from the description of Cl_j similarly to previous case.
- Let us suppose that both clusters are described by several descriptions, $d_{i1}...d_{im}$ and $d_{j1}...d_{jn}$, respectively. If d_{ik} is more general than d_{jh} , then d_{jh} is rejected as in the previous case. There is a dual situation when d_{jh} is more general than d_{ik} .

The result of LazyCL is a set of clusters described by discriminant descriptions that the domain expert can analyze. The best situation about clusters build by LazyCL is when they have empty intersection, since it means that all the known cases can be clearly separated. An interesting situation happens when some clusters have non-empty intersection, although their respective descriptions are not related. This situation can be interpreted, for example, in medical domains as a patient holding a set of symptoms coherent with more than one disease. From a general point of view, this is an opportunity for the expert to discover relationships among classes. For instance, depending on the domain, a possible interpretation of this situation could be to decide that clusters with non-empty intersection are actually the same cluster.

3. Experiments

To analyze the feasibility of LazyCL, we used several data sets from the UCI data set repository (<http://archive.ics.uci.edu/ml/>). Most of them have attributes with numeric values, therefore we discretized them to obtain nominal values. The goal of the experiments

was, on the one hand, to analyze whether or not the clustering is coherent with the correct classification of cases. On the other hand, because the purpose of LazyCL is for knowledge discovery, we also want to analyze the descriptions of the clusters. These descriptions should be understandable enough from the expert's point of view to allow the reconsideration of some clusters (for instance, those with non-empty intersection).

For each data set two different experiments have been carried out: one of them is by generating a random number of clusters as indicated in Section 2; the second kind of experiments is by generating the same number of clusters as classes has the data set. For instance, the *Hepatitis* data set has cases of two classes, *die* and *live*, therefore we imposed the generation of only two clusters. The cardinality of each cluster has to be approximately the same ($Card(Cl_i) = Card(C)/3$, where C is the training set). The idea of this second experiment was to deal with high entropy of clusters, since the situation is that there are the correct number of clusters but they contain elements of all the correct classes. For both kinds of experiments we carried out several executions. Notice that each execution use different clusters since they are randomly created.

Next sections show some interesting results classified in three different situations detected from the analysis of the final clusters.

3.1. Example 1: Clusters with empty intersection

The *Bal* dataset is composed by 625 objects of three classes. The random creation of clusters produced more than 150 clusters in all the experiments. However the result of LazyCL always has been a set of 16 clusters (see some of these clusters in Fig. 2). These clusters have discriminatory descriptions and there is no cases belonging to more than one cluster. We also performed experiments fixing the number of clusters. Thus, because the *Bal* data set has objects of three classes, three clusters with elements randomly chosen have been build. LazyCL produced only one cluster but this cluster is described by 16

```

CLUSTER : CLUST-669 (73 objects)
* (Left-Weight > 2.5) and (Left-Distance > 2.5) and (Right-Weight > 2.5) and (Right-Distance > 2.5)
CLUSTER : CLUST-668 (51 objects)
* (Left-Weight > 2.5) and (Left-Distance > 2.5) and (Right-Weight > 2.5) and (Right-Distance < 2.5)
CLUSTER : CLUST-667 (49 objects)
* ((Left-Weight > 2.5) and (Left-Distance > 2.5) and (Right-Weight < 2.5) and (Right-Distance > 2.5)
CLUSTER : CLUST-666 (33 objects)
* ((Left-Weight > 2.5) and (Left-Distance > 2.5) and (Right-Weight < 2.5) and (Right-Distance < 2.5)
CLUSTER : CLUST-665 (47 objects)
* ((Left-Weight > 2.5) and (Left-Distance < 2.5) and (Right-Weight > 2.5) and (Right-Distance > 2.5)
CLUSTER : CLUST-664 (33 objects)
* ((Left-Weight > 2.5) and (Left-Distance < 2.5) and (Right-Weight > 2.5) and (Right-Distance < 2.5)

```

Figure 2. An sketch of the clustering produced by LazyCL on the *Bal* data set.

CLUSTER : CLUST-109

* D1 : (A03 > 0.287) and (A05 in [0.23, 0.999]) and (A09 < 0.071) and (A11 < 0.583) and (A14 < 0.583) and (A21 < 0.1854) and (A30 < 0.930)

11 objects : (Obj-8 Obj-46 Obj-64 Obj-81 Obj-107 Obj-114 Obj-155 Obj-193 Obj-197 Obj-237 Obj-273)

* D2 : (A03 > 0.287) and (A05 in [0.23, 0.999]) and (A09 < 0.071) and (A14 < 0.583) and (A21 < 0.1854) and (A30 < 0.930) and (A31 in [0.212, 0.986])

11 objects : (Obj-8 Obj-46 Obj-64 Obj-81 Obj-107 Obj-114 Obj-155 Obj-193 Obj-197 Obj-237 Obj-273)

Figure 3. Examples of clusters, for the *Ionosphere* data set, with an inclusion relation of their associated sets.

descriptions, the same ones that the obtained in the previous experiments with random number of initial clusters. We compared these descriptions with the rules obtained from a decision tree constructed using the ID3 algorithm. This decision tree has 16 branches described by exactly the same rules we obtained using LID.

Similar results have been obtained on *Iris* although in this data set the attributes involved in the descriptions of clusters are not exactly the same than the ones used in the decision tree.

3.2. Example 2 : Cluster with descriptions satisfied by the same subset of objects

Despite of the discriminatory definition of cluster descriptions, a possible situation is that the same subset of cases satisfies two descriptions of different clusters. This is not a contradiction but can be interpreted as two views of the same set of objects. An example of this situation can be seen in cluster *Clust-109* from the *Ionosphere* data set (Fig. 3). The two descriptions of this cluster are satisfied by the same subset of examples, the difference between both descriptions is that *D1* has the attribute ($A11 \leq 0.583$) and the description *D2* has the attribute ($A31 \in [0.212, 0.986]$). This is an opportunity for knowledge discovery, since domain expert could analyze both descriptions and to decide that both attributes, *A11* and *A31* are irrelevant. Therefore the description for the cluster *Clust-109* could be the following: ($A03 \geq 0.287$) and ($A05 \in [0.23, 0.999]$) and ($A09 \leq 0.071$) and ($A14 \leq 0.583$) and ($A21 \geq 0.1854$) and ($A30 \leq 0.930$).

3.3. Example 3 : Clusters with non-empty intersection

This situation is similar to the described in previous section. Given two clusters Cl_i and Cl_j described respectively by d_i and d_j , let S_i be the set of cases that satisfy d_i and S_j the set of cases that satisfy d_j . If $S_i \subset S_j$ then the expert can analyze both descriptions to decide about the relevance of the attributes that both description do not share. As before, if differences are found irrelevant, both clusters can be merged.

An example of this situation are clusters *Clust-118* and *Clust-103* of the *Hepatitis* data set. All the cases in the cluster *Clust-118* are also in *Clust-103*, except *obj-3*, although there are several differences among the descriptions of both clusters (see Fig. 4). Here, the domain expert can decide whether or not the shared attributes are the relevant ones to describe a unique cluster. Another possibility is that differences among descriptions be actually important to define two clusters. In such situation, the cases satisfying

CLUSTER : CLUST-118

* ((Steroid (Yes)) (Antivirals (No)) (Fatigue (No)) (Alk_Phosphate (All)) (Histology (No)))

11 objects : (Obj-3 Obj-4 Obj-58 Obj-83 Obj-91 Obj-92 Obj-108 Obj-115 Obj-123 Obj-125 Obj-139)

CLUSTER : CLUST-103

* ((Steroid (Yes)) (Antivirals (No)) (Fatigue (No)) (Spiders (No))
(Albumin (More-Than)) (Histology (No)))

13 objects : (Obj-4 Obj-10 Obj-58 Obj-65 Obj-83 Obj-84 Obj-91 Obj-92 Obj-108 Obj-115 Obj-123 Obj-125 Obj-139)

Figure 4. Example of clusters obtained for the *Hepatitis* data set, where the subset of cases that satisfy the descriptions of the clusters do not have an inclusion relation between them.

both descriptions could be interpreted as having two possible classifications depending on the attributes that are taking into account (i.e. different views of the same objects).

3.4. Discussion

Experimentation consisted on two kinds of experiments: (a) those where both the initial number of clusters and the number of cases inside each cluster have been chosen randomly; and (b) those where the number of clusters was fixed to the number of classes of the data set and the content of clusters is randomly chosen. The result of both kinds of experiments is equivalent. This is surprising since in experiment (b) clusters have high entropy and, in consequence with this fact, we expected that the clustering process finishes with non-discriminant descriptions. Nevertheless, the experiments produced cluster descriptions that are discriminant in both kinds of experiments. Thus, for domains such as *Bal* and *Tao*, both experiments result in the same descriptions and, even more, these descriptions were the same obtained by a decision tree. For other domains, probably less regular than the two mentioned before, clusters are also well formed although the descriptions are more specific than the obtained by a decision tree. As we analyzed in [2] this is an expected result because descriptions from a lazy learning method describe only an area around the problem that is being solved. Differences among experiments on the same data set come from the descriptions of clusters. This fact was expected because relevant attributes depend on the correct partition, and this is different in each experiment.

Concerning the explanations, we confirmed the feasibility of using explained case-based reasoning for knowledge discovery. Explanations provided by LID are easily understood by domain experts since these descriptions both involve the same attributes than the used to represent domain objects and they are discriminant.

LazyCL does not need to choose the number of desired clusters as happens in most of the clustering approaches since the process seems to converge to a correct clustering independently of the initial number. In fact, this is an interesting aspect to analyze in the future, since this result is surprising due to the dependence that LID has on the correct partition. Let us explain this dependence in more detail.

LID uses an heuristic measure based on the information gain (the López de Mántaras' distance [16]) that compares the partition induced by a particular attribute with the correct partition, which clearly plays a crucial role since, depending on it relevant

attributes could be different. Nevertheless, in LazyCL, where the clusters and the cases contained on these clusters have been randomly selected, the correct partition is not actually the “correct” one. Despite of this, the final clustering coincides with the actual classes of the data set (at least for some of the data sets we used in our experiments). From a theoretical point of view, we search for a formal justification of the convergence of LazyCL to a correct clustering. The goal is to prove why from an initial random clustering and, by using generalizations and relations among them, the process always converges to the correct clustering.

4. Related Work

Most common techniques used for knowledge discovery are clustering methods. In [6] the reader can find a survey of such methods and a classification of them. Differently than the LID method that deals with symbolic data, most of clustering methods work better with numerical data since they have their roots on statistics. These numerical methods group objects taking into account both the similarity among the objects included in a cluster and the dissimilarity among objects of different clusters. A different approach is the one taken by *conceptual clustering* where the goal is to build a compact and understandable concept description for each one of the clusters. Commonly, this description is represented by means of a conjunctive logical expression. Clearly, LazyCL can be classified as a conceptual clustering method, since cluster descriptions are oriented to be understood by a human expert. One relevant difference between LazyCL and other conceptual clustering methods is that some methods begin with random seeds from which they construct a concept hierarchy. Instead LazyCL begins with a completely random clustering of data set and the final clusters can be non-disjoint. In fact, this random initialization could be related with clustering methods such as Self-Organizing Maps [14] or K-means [13] that beginning with random seeds they converge to a correct clustering.

One of the early methods for conceptual clustering is COBWEB [8] that defines a tree whose nodes represents concepts. Each concept is described by a set of attributes and the values of these attributes represent the probability than an object takes a value. Because of the tree structure, the measure of the quality of the tree is assessed using the *category utility* measure [11] that is similar to the *information gain* used in decision trees and also in LID.

CITree [15] builds a decision tree such that its leaves represent the clusters. However, to build a decision tree it is necessary that data belong to some class and, for this reason, previously to the construction of the tree, available data are classified as belonging to a class, say *A*. Then a set of random objects are created and labeled as belonging to another class *B*. The process is based on the assumption that available data (i.e., those objects in *A*) have an uniform distribution, since it is possible to define clusters on them, whereas objects in *B* are created in a non-uniform way. CITree and LazyCL use a measure based on the information gain and both work on artificial clusters. However initial clusters of LazyCL are randomly created without any assumption on the uniformity of data.

5. Conclusions

In the current paper we have proposed LazyCL, a procedure for knowledge discovery based on explanations produced by a lazy learning method. Our approach is based on the hypothesis that, because explanations are generalizations, they can be used as domain theory. Although this domain theory is lazy –hence it does not cover all the space of known examples– it can play the same role that domain theory obtained from eager learning methods. However, the application of supervised learning methods require that cases have a class label. Thus, first LazyCL randomly creates clusters and then uses LID with leave-one-out on them. Then the analysis of relationships among explanations produces an understandable clustering of the dataset.

An interesting point is when these clusters have non-empty intersection, since this situation gives to domain experts a good opportunity for knowledge discovery. Experiments on some UCI repository data sets confirm that LazyCL produces well formed clusters despite of the randomness of the process.

Currently, LazyCL cannot be applied for data mining because of the leave-one-out method is impractical for huge data sets. A way to reduce this cost and make the process scalable could be to select a subset of cases on which apply the leave-one-out. This process should be similar to the divide-and-conquer approach for clustering defined in [12]. This could be an interesting line of future research.

Another direction of future work could be experimenting with lazy learning methods different than LID (for instance with Lazy Decision Trees), in order to make LazyCL independent of the problem solver.

Acknowledgments

This work has been supported by the MCYT-FEDER Projects called MID-CBR (TIN2006-15140-C03-01) and the *Generalitat de Catalunya* under grant 2005-SGR-00093. The author thanks to Àngel García-Cerdàña and Josep Lluís Arcos their helpful comments to improve this work.

References

- [1] E. Armengol. Usages of generalization in cbr. In R.O. Weber and M. M. Richter, editors, *ICCBR-2007. Case-based Reasoning and Development*, number 4626 in Lecture Notes in Artificial Intelligence, pages 31–45. Springer-Verlag, 2007.
- [2] E. Armengol. Building partial domain theories from explanations. *Knowledge Intelligence*, (in press), 2008.
- [3] E. Armengol and E. Plaza. Lazy induction of descriptions for relational case-based learning. In L. De Raedt and P. Flach, editors, *ECML-2001.*, number 2167 in Lecture Notes in Artificial Intelligence, pages 13–24. Springer, 2001.
- [4] E. R. Bareiss, B. W. Porter, and C. C. Wier. Protos: an exemplar-based learning apprentice. *Int. J. Man-Mach. Stud.*, 29(5):549–561, 1988.
- [5] R. Bergmann and A. Stahl. Similarity measures for object-oriented case representations. In *Proc. European Workshop on Case-Based Reasoning, EWCBR-98*, Lecture Notes in Artificial Intelligence, pages 8–13. Springer Verlag, 1998.
- [6] P. Berkhin. Survey of clustering data mining techniques. Technical report, Accrue Software, San Jose, CA, 2002.

- [7] C.L. Blake and C.J. Merz. UCI repository of machine learning databases, 1998.
- [8] D. H. Fisher. Knowledge acquisition via incremental conceptual clustering. *Machine Learning*, 2:139–172, 1987.
- [9] W. J. Frawley, G. Piatetsky-Shapiro, and C. J. Matheus. Knowledge discovery in databases - an overview. *Ai Magazine*, 13:57–70, 1992.
- [10] J. H. Friedman, R. Kohavi, and Y. Yun. Lazy decision trees. In *AAAI/IAAI, Vol. 1*, pages 717–724, 1996.
- [11] M. A. Gluck and J. E. Corter. Information, uncertainty, and the utility of categories. In *Program of the Seventh Annual Conference of the Cognitive Science Society*, pages 283–287, 1985.
- [12] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Computing Surveys*, 31(3):264–323, 1999.
- [13] T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, and A.Y. Wu. The analysis of a simple k-means clustering algorithm. In *Symposium on Computational Geometry*, pages 100–109, 2000.
- [14] T. Kohonen. The self-organizing map. In *Proceedings of the IEEE*, 78:1464–1480, 1990.
- [15] Bing Liu, Yiyuan Xia, and Philip S. Yu. Clustering through decision tree construction. In *CIKM '00: Proceedings of the ninth international conference on Information and knowledge management*, pages 20–29, New York, NY, USA, 2000. ACM.
- [16] R. López de Mántaras. A distance-based attribute selection measure for decision tree induction. *Machine Learning*, 6:81–92, 1991.
- [17] T. M. Mitchell, R. M. Keller, and S. T. Kedar-Cabelli. Explanation-based learning: A unifying view. *Machine Learning*, 1(1):47–80, 1986.
- [18] E. Plaza, E. Armengol, and S. Ontañón. The explanatory power of symbolic similarity in case-based reasoning. *Artificial Intelligence Review. Special Issue on Explanation in CBR*, 24:145–161, 2005.
- [19] J. R. Quinlan. Induction of decision trees. *Mach. Learn.*, 1(1):81–106, 1986.

Improving Pseudobagging techniques

Angela Chieppa^{a,1}, Karina Gibert^a, Ignasi Gómez-Sebastià^b, Miquel Sànchez-Marrè^b

^a*Statistics and Operations Research Department*

^b*Computer Software Department*

Technical University of Catalonia (UPC), Barcelona

Abstract. We present an important improvement related to the computation and use of Mutual Information index in Pseudobagging, a technique that adapts “bagging” to unsupervised context. The Mutual Information index plays a key role in this technique, assessing the quality of a partition. We propose the use of such an index to improve the Pseudobagging voting scheme for determining the final partition of the data. Issues related to the estimation of Mutual Information index in the multivariate continuous case become crucial for the application of Pseudobagging to real data: we discuss some practical approaches to computation in this situation. Finally, experimental results are presented, related to application of new “pooled voting” scheme and to the evaluation of the impact of different computing methods for Mutual Information.

Keywords. cluster ensemble, discretization, mutual information, pooled voting.

1. Introduction

Class discovery in an unknown domain is the goal of clustering algorithms, which usually result in partitioning the observational dataset: for many years efforts have been concentrated upon improving single algorithms, but recently this problem has been addressed in a different way [1][5]: combining the results of multiple clusterings in order to obtain “robust” data partitions, as supervised *ensemble techniques* such as bagging [1] do.

Formally, given N different available partitions of the data X , a *clustering ensemble* $\Pi = \{P_1, P_2, \dots, P_N\}$, is defined, where $P_i = \{C_{i1}, C_{i2}, \dots, C_{i\xi_i}\}$ is a single partition with ξ_i clusters.

The goal consists in defining a final “consensus” partition P^* , result of a combination of the N partitions of Π . Determination of P^* always involves the use of a “quality” measure for partitions: authors agree on the use of the Mutual Information index as a measure of how well final partition reproduces information in original data [5] or shared in cluster ensemble [1].

Pseudobagging technique [5] is one of these cluster ensemble techniques: one of the specificities of Pseudobagging is the use of Mutual Information and Inertia as quality measures, useful both either in an intermediate step of the Pseudobagging process and for the assessment of goodness of the final results.

¹ Statistics and Operations Research Department, Edifici C5 Campus Nord UPC, carrer Jordi Girona 1-3, 08034 Barcelona; email: angela.chieppa@gmail.com

Computation of Mutual Information index requires the estimation from observed data of joint density of all the attributes considered: in the multivariate continuous case, where numerical integration is required, this involves some computational handicaps. These issues have been addressed in several works [6], [9]: in this paper, the computation of the index on discretized data is proposed, as other interesting methods still are very difficult to apply when dealing with more than a few variables.

Experimental results related to robustness of Mutual Information index using different discretization methods are presented. Tests have been run using a set of data coming from a waste water treatment plant [4] and Data Analysis Intelligent System GESCONDA [4], which has been specifically designed and developed at *Technical University of Catalonia (UPC)* and implements Pseudobagging, among others.

The paper starts describing clustering ensemble methods and Pseudobagging. Then questions on Mutual Information computation with numerical attributes are addressed, and methods chosen for discretization are described. A description of how MI could be used for weighting voting follows. Finally, experimental results are presented. The paper ends with some conclusions and future work.

2. Cluster Ensemble Methods and Pseudobagging

Cluster ensemble methods are the “unsupervised version” of more general multi-classifiers systems, that improve the accuracy of a learning algorithm by conveniently “combining” the results of a multiple or recursive application of it [1].

In unsupervised framework, the lack of knowledge about *true* class labels makes impossible to express a prediction error and consequently a loss function, that along with its minimization allows increasing accuracy in final classification and comparing the effectiveness of different methods: so some “criteria” or new objective function to evaluate final partition has to be defined. In clustering ensemble methods these proposed criteria take into account two main properties: first of all, goodness of fit to the original observed data, the more the partition resembles the original data, the better; then specificity of problem suggest also to consider the *consistency* with the clustering ensemble, involving the *agreement* between P^* and single P_i partitions in the ensemble (P^* as a *consensus* partition).

Recent works [1],[11] agree in the definition of the *normalized average mutual information* of a partition as the objective function to be maximized: this function is a measure of the shared information of a single partition with all the others, according to the Mutual Information index. However, this function is impractical for direct optimization so proposed techniques mostly use empirical validation and heuristics.

There are different cluster ensemble techniques using Mutual Information to assess quality of partition. Fred and Jain [1] introduce the concept of evidence accumulation clustering that maps the individual data partitions in a clustering ensemble into a new similarity measure between patterns, summarizing inter-pattern structure among clusterings; a final data partition is obtained by applying the single-link method to the new similarity matrix. Strehl and Ghosh [11] explore the concept of consensus between data partitions, using graph-theoretical approaches for consensus decisions, based on a cluster matching paradigm.

Pseudobagging [5] adapts classical bagging supervised technique to an unsupervised context: partitions are obtained making use of repeated runs of k-means (notice that other partitioning techniques are possible), with random initial seeds.

Randomizer effects, that could lead to partitions of very different qualities, are *mitigated* by combining partitions; a simple voting scheme is used to combine different partition results, each partition voting with a single vote. In the unsupervised context, combining labels from several iterations requires a pre-process of relabelling all partitions with regards to a *reference* one, since the labels are automatically produced in the clustering process and the same class may change label from one iteration to another (making differences only *apparent*): that's why before applying voting, best partition is chosen, on the basis of Mutual Information Index or Inertia, and labels of this partition are used to re-label correspondent ones in other partition to be combined. Finally, quality of resulting consensus partition is judged, with MI or Inertia again.

3. First implementation of Pseudobagging in GESCONDA

Pseudobagging technique has been implemented in GESCONDA, an Intelligent Data Analysis System, designed and developed at Technical University of Catalonia (UPC) [2], for knowledge management mostly in environmental databases. This system has been developed in Java and integrates both statistical and artificial intelligence data mining techniques.

First implementation of Pseudobagging was very useful to test main features of algorithm [5]: increased quality of *bagged* final classification and results became stable using just a few partitions to combine. This first implementation, however, presented some drawbacks. First, it used a simple majority voting scheme, that is to say that each partition has the same weight when voting, which is quite inefficient, taking into account that different partitions have different quality that is already taken into account in re-labelling step of the Pseudobagging process. Moreover, the computation of the MI index on previously discretized continuous attributes can change depending on the underlying discretization method used. Following paragraphs address these issues and present methodological solutions chosen to improve the quality of the proposal.

4. Computation of Mutual Information with Continuous Attributes

Original MI index was proposed to measure distance from independence between two variables: extension to multivariate case is intended as a measure of simultaneous interaction or a sort of a multi-way similarity: if 0, variables “do not interact” [9]. Formally, given X_1, \dots, X_K variables, the mutual information MI is calculated as:

$$MI(X_1 \dots X_K) = \int_{-\infty}^{+\infty} \dots \int_{-\infty}^{+\infty} f(x_1 \dots x_K) \log \frac{f(x_1 \dots x_K)}{(f_1(x_1) \dots f_K(x_K))} dx_1 \dots dx_K$$

where $f(x_1 \dots x_K)$ represents the observed joint density function between X_1, \dots, X_K and each $f_k(x_k)$ the univariate density function of X_k , with $k=1, 2, \dots, K$. When the variables are categorical, it is possible to add the frequencies of the modalities of those variables as estimations of the probabilities. In this case, every X_k can take a set of possible discrete value $D_k = \{x_{k1} \dots x_{kt}\}$ and the formula becomes:

$$MI(X_1 \dots X_K) = \sum_{x_1 \in D_1} \dots \sum_{x_K \in D_K} p(x_1 \dots x_K) \log \frac{p(x_1 \dots x_K)}{(p_1(x_1) \dots p_K(x_K))}$$

where $p(x_k) = P(X_k = x_k)$, probability approximated by observed relative frequency of x_k in the dataset and $p(x_1, x_2, \dots, x_K)$ is the joint probability of “ $X_1 = x_1$ and $X_2 = x_2 \dots$ and $X_K = x_K$ ” approximated similarly.

Problems in computation of MI arise when working with continuous variables: how to estimate the joint density function? Many methods have been developed to solve this density estimation problem, using a kernel density approach or a segmentation of variables spaces [9][12][7][6] : a completely satisfactory method to deal the question when a big number of numerical attributes is involved is difficult to find or, at least, very difficult to implement. Managing this computation by previously discretizing numerical attributes seems to be a good solution from a practical point of view, as long as discretization process doesn't imply relevant information loss: that's why it's important to evaluate the “robustness” of MI index with respect to different discretization methods used.

There are a lot of different methods [8], we concentrate on three methods: *equidistant* method, for its simplicity and diffusion could be considered a baseline; *boxplot-based discretization* method [4], which has given very good results and for this reason already was implemented in GESCONDA; an *entropy-based* discretization method, using Fayad-Irani algorithm [10] [8].

4.1. Equidistant discretization

Equal width interval binning is perhaps the simplest method to discretize data and surely one of the most used. It divides, after sorting observed values, the range of the variable to be discretized into l equally sized bins, where l is a parameter introduced by user. Formally, if a variable X has boundary values x_{\min} and x_{\max} , then discretization is obtained determining first of all the bin width $\delta = (x_{\max} - x_{\min})/l$, and then constructs cutpoints at $(x_{\min} + i\delta)$, where $i = 1, \dots, l-1$. Its main drawback is “sensitivity” to outliers, that may drastically skew the range; also, this may produce disequilibrium in the sense of producing intervals with very different frequencies.

4.2. Boxplot-based discretization

The *Boxplot based discretization* is presented in [3]: it's a supervised method which uses the same idea of boxplot graphical representation to induce cutpoints for partitioning numerical attributes on the basis of a previous classification. Given a partition P with ξ classes, method consists in:

1for each class C_j , $j = 1, \dots, \xi$, computing minimum (m_k) and maximum (M_k) value for X_k , numerical variable to be discretized; a set M with all these values is obtained $M = \{m_1, M_1, \dots, m_\xi, M_\xi\}$

2determining set $Z = \{z_1, z_2, \dots, z_{2\xi}\}$ of cutpoints by sorting M

The bins obtained do not have the same length and the set of intersecting classes is constant all along every interval and changes from one to another. An interesting

property of this method is that it produces a discrete variable which has maximum association with partition P.

4.2.1. Fayyad-Irani Entropy Discretization

This supervised algorithm is a generalization of a binary discretization algorithm based on the minimization of information entropy, calculated on the basis of the class labels: it finds the best split so that the bins are as pure as possible, i. e. the majority of the values in a bin have the same class label. For details, better referring to the original article [10]. The core binary discretization technique is based on determining cutpoint T by the minimization of the *class information entropy of partition induced by T*:

$$E(X_k, T, S) = \frac{|S_1|}{|S|} Ent(S_1) + \frac{|S_2|}{|S|} Ent(S_2)$$

where S is the set of instances in observed dataset, S₁ is the subset including all instances with a lower value of X_k than T and S₂ = S - S₁; finally, Ent(S_i) is the entropy of a subset S_i with reference classification.

The process is recursively applied to partitions obtained until the stopping criterion is met. The use of the “Minimal Description Length Principle” to decide when stop splitting is the novel idea presented by the authors of this algorithm.

5. Determining final partition in Pseudobagging: the voting scheme

In the first implementation of Pseudobagging process [5], the final class is assigned based on a most frequent class according to a basic count: every partition P_i in Π, gets one vote, regardless its quality and the most voted class is the one assigned. This can drive to a biased Pseudobagging class assignation that does not correspond with the reality.

A pooled voting scheme is clearly better: quality index, either the Inertia or MI, already computed for Pseudobagging during relabelling phase, could be used to assign a number of votes to each partition. Two factors are considered:

- Normalized Index: this factor reflects how much the current execution is contributing to the total sum of quality Indexes, in a percentage coefficient.
- Number of distributed votes: this factor reflects the total number of votes to distribute among all partitions. Higher number of votes results in higher precision. Each partition is assigned a number of votes according to normalized index coefficient, that is to say in the same proportion to total number as its quality is providing to total quality: thus, if a given execution is contributing the double than other to total quality, it will get a double number of twice as much votes.

6. Application

Data analyzed in this paper comes from the wastewater Treatment Plant (WWTP) of Girona, Spain. It is a sample of 396 observations taken from September the first of 1995 to September the 30th of 1996 [4]. Each observation refers to a daily mean: the

state of the Plant is described through a set of 25 variables, considered the more relevant upon expert’s opinions. We completely relied on GESCONDA for experiments: version of system being the one with new features implementing pooled voting and discretization methods.

Two main experiments were executed: one to test quality increase of final partition determined by Pseudobagging when using pooled voting instead that simple majority scheme; second test was to prove robustness of MI and of the overall Pseudobagging process towards different computation methods, e.g. different type of discretization of numerical attributes.

Experiments design was strongly affected by results of previous works on the same data [4][5], namely that four is the right number of classes for these data and that ten partitions are enough to get a good classification on these data with Pseudobagging.

To set up both experiments, 20 executions of K-means on WWTP data were performed (P_1, \dots, P_{20}), every execution providing a partition of the 396 days in 4 classes. Then, using increasing sequence of these partitions, 20 cluster ensemble were built $\Pi_1=(P_1), \Pi_2=(P_1,P_2) \dots \Pi_{20}=(P_1,\dots P_{20})$.

6.1.Evaluating pooled voting

For first experiment on evaluation of pooled voting, Pseudobagging was applied to the 20 cluster ensembles $\Pi_1=(P_1), \Pi_2=(P_1,P_2) \dots \Pi_{20}=(P_1,\dots P_{20})$. Firstly, a simple voting scheme was used to find a consensus partition for each of them; then, a pooled voting scheme was used, producing a new series of 20 partitions. So, at the end, 3 series of partitions were available: one including the single k-means executions and other two series with “consensus” partitions resulting from Pseudobagging with the two different voting scheme.

For each Π_i partition in one of these three series and considering the K attributes in the dataset, the Mutual Information Index $MI(X_1,\dots X_K,\Pi_i)$ was computed, using equidistant discretization technique for numerical attributes, as a quality measure.

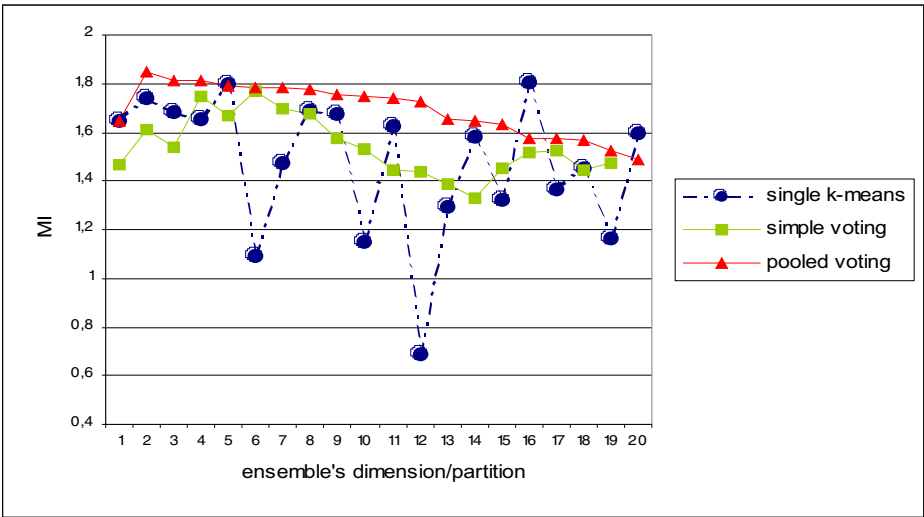


Figure 1. Quality of final partition according to voting scheme

Results (Figure 1) clearly show very good performance of pooled voting, with MI indexes always higher than the ones obtained with simple voting scheme. More, when a “bad” partition enters the ensemble (as partitions 6, 10 and 12) pooled voting is not so strongly affected as simple voting.

In these experimental results, the five initial ensembles considered are composed by higher quality partitions, while low quality ones enter in the following ensembles: this explains the big boost of pooled voting MI series at the beginning and the progressive tendency to go down; when a very good MI partition occurs again, the change in tendency doesn't take place because of the occasionality of the fact and of the dimension of the ensemble, that “dilute” the impact.

6.2. Computing Mutual Information with Continuous Attributes: evaluation of effects of discretization method

Firstly, to evaluate effects of different discretization methods when computing MI index, for each partition P_i of the 20 single k-means executions, $MI(X_1, \dots, X_K, P_i)$ was calculated using different methods, resulting in 3 series of values; for supervised discretization techniques, the same partition P_i was used for determining classes.

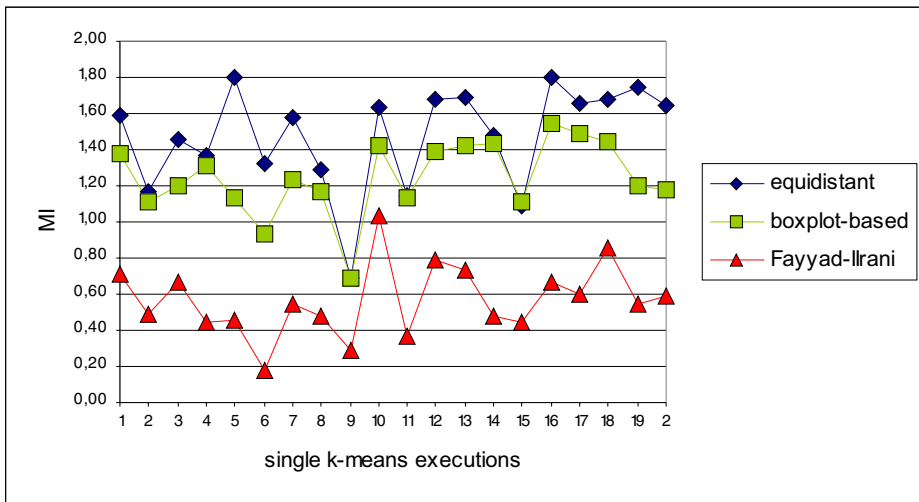


Figure 2 . MI computed using different discretization methods

First thing to remark (Figure 2) is that lower values of MI appear when computing with Fayyad-Irani discretization method. Second remark is the fact that the distribution of the three series is rather similar, that in the Pseudobagging context, where MI index is used to rank partitions of the ensemble, implies that different techniques analyzed produce similar results.

The availability of a reference partition for these data, obtained in previous works [4] and validated by experts, suggested the computation of the “accuracy” of partitions used in the experiment. This accuracy was calculated using a feature of GESCONDA, which provides, among other things, the percentage of correctly classified observations

[2]. So, in order to evaluate the overall effect of discretization of MI in the entire Pseudobagging process, partitions obtained using Pseudobagging on Π_1, \dots, Π_{20} implementing different discretization method were compared through accuracy index.

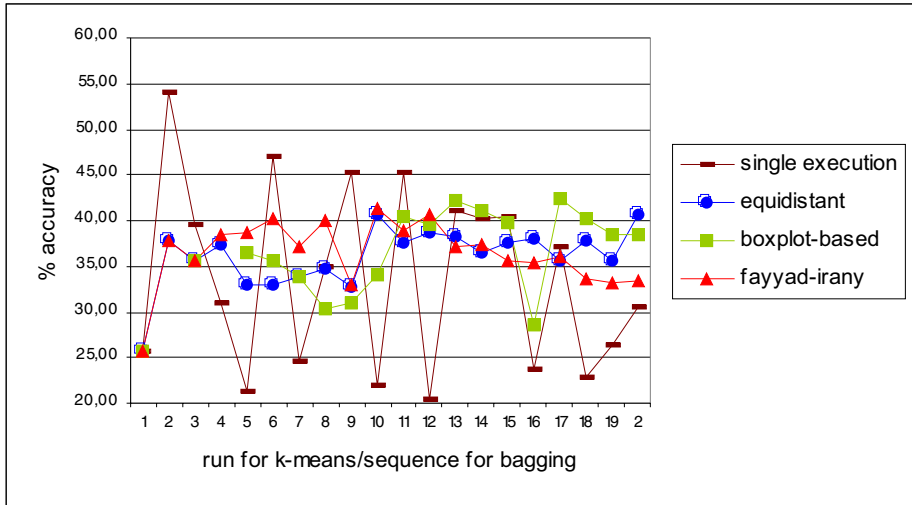


Figure 3. Comparison of Pseudobagging results accuracy using different discretization methods and with single k-means executions

Figure 3 shows first of all good properties of Pseudobagging: high variability in accuracy series of single k-means partitions is contrasted by accuracy of *pseudobagged* partitions, more stable whatever the method. More, there isn't a technique that produces better results than others. So, this experimental results show that in Pseudobagging framework, different discretization methods to compute Mutual Information do not affect final results.

7. Conclusions and future work

In this work special attention is given to Mutual Information index in the context of Cluster ensemble methods: more specifically, in Pseudobagging technique [5]. This technique in its first implementation already made use of MI in a specific step to assess quality of partition: this work proposes the use of the same index also to "weight" voting scheme; results of application to real data prove quality increase in final partition.

Discretization of continuous attributes is one of the main tools used to deal with computational difficulties arising in the continuous multivariate case. Results of application of Pseudobagging techniques using three different discretization methods on data coming from a Waste Water Treatment Plant are discussed, to assess MI computation robustness and effect of different discretization on Pseudobagging results.

Experimental results of this case-study doesn't give strong evidence for a discretization technique against others and show that quality of Pseudobagging is not affected by the choice of the methods for computing Mutual Information index.

Future work will first of all validate results of this work with other datasets; then other interesting ways of computing MI [7] and new possibilities to find final consensus partition will be explored.

References

- [1] Fred, A. L. N.; Jain, A. K.; "Combining Multiple Clusterings using Evidence Accumulation", *IEEE Trans. on Pattern Analysis and Machine Intelligence* 27 (6): pp. 835 - 850, 2005
- [2] Gibert, K.; Sánchez-Marré, M.; "Gesconda: An intelligent data analysis system for knowledge discovery and management in environmental databases", *Environmental Modelling and Software* 21 (1): pp.116-121, 2006
- [3] Gibert, K.; Pérez-Bonilla, A.; "The role of the Boxplot based Discretization in the conceptual interpretation of a hierarchical cluster", Proceedings 'V Taller Nacional de Minería de Datos y Aprendizaje' in: Ferrer-Troyano, F.J.; Troncoso, A., and Riquelme, J.C. (Eds.); *CEDI'07 Zaragoza 12-13 September 2007* : pp. 157-166, Thomson, 2007
- [4] Gibert, K. and Roda, I. "Identifying characteristic situations in wastewater treatment plants", Proceedings of the workshop BESAI in: Horn, W. (ed.): *ECAI 2000, Proceedings of the 14th European Conference Artificial Intelligence, Berlin August 20-25*: vol.1 pp. 1-9, IOS Press, 2000
- [5] Gibert, K.; Pinyol, I.; Oliva, L.; Sánchez-Marré, M. - Pseudobagging: Improving Class Discovery By Adaptive Bagging Techniques To Clustering Algorithms", Proceedings 'V Taller Nacional de Minería de Datos y Aprendizaje' in: Ferrer-Troyano, F.J.; Troncoso, A., and Riquelme, J.C. (Eds.); *CEDI'07 Zaragoza 12-13 September 2007* : pp. 157-166, Thomson, 2007
- [6] Huh, M.; "Subset Selection Algorithm Based On Mutual Information", in: Rizzi, A and Vichi, M. (eds.) *Compstat 2006 - 17th Symposium, Rome: Proceedings in Computational Statistics*, Springer, 2006
- [7] Darbella, G. A.; Vajda, I. "Estimation of the Information by an Adaptive Partitioning of the Observation Space", *IEEE Transaction on Information Theory*, 45 (4), 1999
- [8] Dougherty J.; Kohavi R.; Sahami M., "Supervised And Unsupervised Discretization Of Continuous Features", in: Prieditis, A. and Russell, S. (eds.), *Machine Learning: Proceedings of the Twelfth International Conference*, Morgan Kaufmann Publishers, 1995
- [9] Kojadinovic, I., On the use of Mutual Information in data analysis, in: *Proc. of the 11th international symposium on Applied Stochastic Models and Data Analysis (ASMDA'05)*: pp. 738-747, 2005
- [10] Fayyad U. M.; Irani, K. B., "Multi-Interval Discretization Of Continuous-Valued Attributes For Classification Learning", in Proceedings of the 13th International Joint Conference on Artificial Intelligence, Morgan Kaufmann, 1993
- [11] Strehl and J. Ghosh. "Cluster ensembles - a knowledge reuse framework for combining multiple partitions". *Journal of Machine Learning Research*, 3 (Dec): pp. 583-617, 2002.

Knowledge Discovery on the response to neurorehabilitation treatment of patients with Traumatic Brain Injury through an AI&Stats and graphical hybrid methodology

Karina GIBERT^{a,b,1}, Alejandro GARCÍA-RUDOLPH^c, Alberto GARCÍA-MOLINA^c,
Teresa ROIG-ROVIRA^c, Montserrat BERNABEU^c, José Maria TORMOS^c

^a *Statistics and Operation Research Department. Universitat Politècnica de Catalunya. Barcelona, Spain*

^b *Knowledge Engineering and Machine Learning group, Universitat Politècnica de Catalunya. Barcelona*

^c *Institut Guttmann-Hospital de Neurorehabilitació. Badalona, Spain*

Abstract. *Neuropsychological rehabilitation seeks to reduce cognitive disability after acquired brain injury. However, until now, there is not enough data to allow exercise of neuropsychological rehabilitation based on scientific Class I evidence. From the medical point of view, the purpose of this work is to develop a classificatory tool to identify different populations of patients based on the characteristics of deficit and response to treatment. This Knowledge Discovery problem has been faced by using exogenous clustering based on rules, an hybrid AI and Statistics technique, which combines some Inductive Learning (from AI) with clustering (from Statistics) to extract knowledge from certain complex domains in form of typical profiles. In this paper, the results of applying this technique to a sample of patients with Traumatic Brain Injury are presented and their advantages with regards to other more classical analysis approaches are discussed.*

Keywords: *Decision Support and Knowledge Management, Rehabilitation, clinical test, Traumatic Brain Injury, Knowledge Discovery, exogenous clustering based on rules, Knowledge-based applications in Medicine.*

Introduction

Neuropsychological rehabilitation seeks to reduce cognitive disability after acquired brain injury. However, there is not enough data yet to allow neuropsychological rehabilitation based on Class I evidence (well designed, prospective, randomized

¹ Corresponding Author: Karina Gibert Oliveras, PhD. Statistics and Operation Research. Universitat Politècnica de Catalunya. Edifici C5 Campus Nord. C Jordi Girona, 1-3 08034 Barcelona Tel (+34) 934 017 323 Fax: (+34) 934 015 855
E-mail: karina.gibert@upc.edu

controlled studies). Although there is a considerable amount of comparative studies aimed to show strongest efficacy of rehabilitation versus other interventions, most of them remain inconclusive. The international scientific community sustains the existence of intrinsic characteristics of this population of patients that make difficult the use of the standard methodology used in other therapies or clinical trials. The strongest factors detected are the heterogeneity of the studied populations with a lack of knowledge about the natural evolution of the process for different patients, and the lack of knowledge about the “active” components of the treatments to be controlled. Nowadays it is well known that Knowledge Discovery (KDD) provides a good framework to analyze complex phenomena as the one presented here for getting novel and valid knowledge that can improve the background *doctrine corpus* [1-2]. The aim of this research is to develop a classificatory tool to identify different populations of patients based on the characteristics of deficit and response to treatment. This leads to the identification of distinguishable subpopulations of similar patients by extracting knowledge from the collected database and seeing how the neuropsychological tests scores provide information for identifying typical profiles of patients. This will help to isolate more homogeneous groups to conduct further studies on the comparisons of active elements of the intervention, and determine the influence of premorbid determinants and characteristics of the lesion, the deficit and residual functionality.

So, typical responses to rehabilitation are to be identified, together with the characteristics of the groups of patients who provide each *type* of response. In fact, this raises a clustering problem. Classical clustering techniques cannot well recognize certain domain structures, so producing some non-sense classes, which cannot be interpreted by the experts [3]. Actually, this arises when dealing with *ill-structured domains (ISD)* [4-5], where numerical and qualitative information coexists, and there exists some relevant semantic additional (but partial) knowledge to be regarded. *Clustering based on rules (CIBR)* [4] was introduced by Gibert to improve clustering results on *ISD*. In fact, one of its main advantages is that it guarantees the semantic meaning of the resulting classes. In previous works it has been discussed the improvements in results related with *CIBR* instead of using other classical clustering techniques. In this work, *CIBR* is extended to a more general methodology where interpretation support tools are also considered.

Paper starts with an introduction to the neuropsychological tests used, description of the target sample and the characteristics of the study, followed by details about the methodology. Afterwards, results of applying *ECIBR* to the sample and use of Class Panel Graph [6] to interpret the results. Finally, discussion and conclusions are provided.

1. Methods

1.1. Neuropsychological tests

All patients meet criteria to initiate neuropsychological rehabilitation. Neuropsychological assessment (38 variables) covered the major cognitive domains (language, attention, memory and learning, and executive functions). Language tests assess repetition of words, confrontation naming, and verbal comprehension. Measures of attention included Digit Span Forward, Trail Making Test-A, Sustained Attention

Test, and Stroop Test (word-colour condition). Memory and Learning was assessed with Digit Span Backward, Immediate and delayed stories from the PIEN and Learning Curve Test. Executive functions was assessed with the Wisconsin Card Sorting Test, Trail Making Test-B, and Stroop Test (interference condition)[7]. After the initial evaluation all the patients initiated a two months program with a personalized intervention based on rules, where patients worked in each one of the specific cognitive domains, considering the degree of the deficit and the residual functional capacity.

1.2. Experimental procedure

The target sample includes 47 patients with moderate or severe Traumatic Brain Injury (TBI) between 17 and 68 years, receiving neurorehabilitation treatment at the *Institut Guttmann-Hospital de Neurorehabilitació* from November 2005 to December 2006. All patients were administered the neuropsychological assessment at admission. Same evaluation was also performed at the end of the rehabilitation. Differences between pre- and post-treatment test scores were used to measure particular patient's improvements in the domains of language, attention, memory and executive functions.

1.3. Data analysis methodology

A brief description of the whole proposed analysis methodology is presented. First, *descriptive statistics of every variable* was done. Very simple statistical techniques [8] were used to describe data and to get preliminary information about: histograms or bar charts to display variability, plots and multiple box-plots to observe the relationship between some pairs of variables, etc; classical summary statistics were also calculated. Next, *data cleaning*, including missing data treatment or outlier detection was performed. It is a very important phase, since the quality of final results directly depends on it. Decisions were taken on the basis of descriptive statistics and background knowledge of the experts. A selection of relevant variables was also done and it was decided that only measures before the treatment and differences will be used in the analysis, while measures after the treatment will only be used to enrich final interpretation.

Data was analyzed using a generalization of *Clustering based on rules (CIBR)*, described below, with Ward's criteria and *Gibert's mixed metrics* [6] (see [4-5] for details). *CIBR* is a hybrid AI and Statistics technique which combines inductive learning (AI) and clustering (Statistics) specially designed for KDD in *ISD*, where a *Knowledge Base (KB)* is considered to properly bias the clustering. Implemented in the software *KLASS* it has been successfully used previously. Our experience [9, 3, 10-13] shows that *CIBR* outperforms classical statistical clustering method by itself, since an important property of the method is that semantic constraints implied by the *KB* are hold in final clusters; guaranteeing *interpretability* of the resulting classes. Also, it is usually better than pure inductive learning methods, since it reduces the effects of missing some implicit knowledge in the *KB*. The general idea is:

1. Build a *Knowledge Base (KB)* with additional prior knowledge provided by the expert, which can even be a *partial* description of the domain.
2. Evaluate the *KB* on data for *inductive learning* of an initial partition on part of the data; put the data not included in this partition into the *residual class (RC)*.

3. Perform one independent hierarchical clustering for every *rules-induced class (RIC)*.
4. Generate prototypes of each *rules-induced class*.
5. Build the *extended residual class* as the union of *RC* with the set of prototypes of *RIC*, conveniently weighted by the number of objects they represent.
6. Perform a weighted hierarchical clustering of the *extended residual class*.
7. In the resulting dendrogram, substitute every rules-induced prototype by its hierarchical structure, obtained in 3. This integrated all the objects in a single hierarchy.

In this work a generalization of *CIBR* is used, namely *Exogenous Clustering based on rules (ECIBR)*, which consists on providing the possibility of defining the *KB* in terms of variables which will not be considered in the clustering process itself. This allows the experts to express their prior knowledge even using complex concepts that can eventually integrate several variables in a single one. In this particular case, the expert knowledge is expressed in terms of pre or post measurements and differences, while clustering is using pre measurement and differences, but not post measurements. Results are, graphically represented in a *dendrogram* (a binary tree where the height of nodes indicates the homogeneity of the class). Final number of classes was determined on best horizontal cut (where the largest branches gap exists, equivalent to maximize the ratio of between-classes inertia vs within-classes inertia). This identifies a partition of the data.

In this work, inclusion of some postprocessing techniques in the global methodology, oriented to support the interpretation, is proposed. Thus, interpretation of the classes was based on *Class Panel Graph (CPG)* [6], where conditional distributions of the variables through the classes, via multiple histograms, is shown in a compact way. The corresponding significance tests were used to assess relevance of differences (ANOVA, Kruskal-Wallis or χ^2 independence test, depending on the item), so it permits to identify which are the variables characterizing the different classes. The aim is to extract qualitative information from the *CPG*, to obtain a meaningful description of classes by identifying which variables indicate particularities of every class regarding the others. A final proposed methodology for Knowledge Discovery using *ECIBR* and *CPG* is:

1. Build a *Knowledge Base (KB)* with additional prior knowledge provided by the expert, which can even be a *partial* description of the domain.
2. Evaluate the *KB* on data for *inductive learning* of an initial partition on part of the data; put the data not included in this partition into the *residual class (RC)*.
3. Select the set of active variables, to be used in the clustering.
4. Perform one independent hierarchical clustering for every *rules-induced class (RIC)* with the active variables.
5. Generate prototypes of each *rules-induced class* (using only active variables).
6. Build the *extended residual class* as the union of *RC* with the set of prototypes of *RIC*, conveniently weighted by the number of objects they represent.
7. Perform a weighted hierarchical clustering of the *extended residual class* with the active variables.

8. In the resulting dendrogram, substitute every rules-induced prototype by its hierarchical structure, obtained in 3. This integrated all the objects in a single hierarchy.
9. Select the number of final classes upon the dendrogram.
10. Obtain the final partition with the selected set of classes (identify the objects belonging to every class).
11. Use the final partition with the complete set of variables to perform a Class Panel Graph.
12. Identify the characteristic variables of every class for interpretation.

2. Results

Forty-seven persons with TBI (41 men, 6 women) participated in the study. The mean age at time of injury was 28.36 years (SD=12 years). Their initial clinical severity of TBI according to Glasgow Coma Scale (GCS) was estimated as severe (GCS between 3 and 8) in 34 (71.43 %) and moderate (GCS between 9 and 12) in 13 (28.57%) cases. First, a classical hierarchical clustering [14] was applied without considering the KB with the prior expert knowledge. Only a trivial distinction between respondents and non-respondents was obtained. In fact, 4 classes were suggested, three of them for non-respondents but very confusing and without clear differences between them. Using *ECIBR*, 5 classes (and one outliers) with different responses were found. A clear conceptual interpretation is possible from the experts' point of view, looking at different variables [9]. Part of the CPG supporting this interpretation is in Figure 1:

- **Assessable (V14)**, including patients with mild to moderate impairment on the administered evaluation, which generally improves after the treatment up to normality in many cases. This is the only group which could be assessed at the beginning of the treatment.
- **Global Improvement (I11)**, with severe impairment before treatment and global satisfactory improvement after it, reaching in many cases scores similar to Assessable group (V14). Indeed this is the group which experiments greater improvement regarding their initial conditions and the final ones.
- **Dysexecutive (I12)**, with severe impairment before treatment and satisfactory improvement in attention, memory and learning. However they show persisting executive functions disorder, remaining unable to develop complex routines.
- **Resistant (G9)**, with severe impairment before treatment and mild improvement in tests requiring minimal attention, quite consistent improvement in memory and learning skills; but remaining deficit for attention complex tests and executive functions measures, being dependent for daily live activities. This group is the one with lower response to treatment.
- **Language (G12)**, including patients with language problems and very severe global cognitive impairment that only improve in correlation with language recovery.

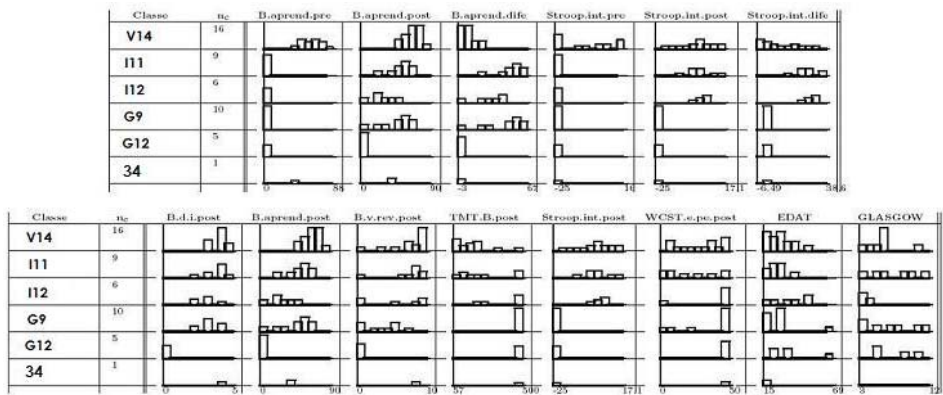


Figure 1. Class Panel Graph of most relevant variables. The *Classe* column shows the identified profiles (classes, labeled V14,I11,I12,G9,G12 and the outsider 34, before interpretation); n_c notes the number of individuals in each class. One histogram is presented for each of the selected variables in each identified class. The first one (*B.aprend.pre*) shows that only individuals in class V14 obtained results above zero. In the next one (*B.aprend.post*) an improvement in performance is shown in every class, except for G12. The difference between them is also shown in the third one (*B.aprend.dife*). Following columns show other selected variables. The last two histograms (*EDAT* and *GLASGOW*) show the patients age and injury severity for each of the identified classes.

3. Discussion and conclusions

Clustering techniques allows detecting different groups of patients using their neuropsychological evaluations. The analysis of the data under classical hierarchical clustering only provides a trivial distinction between patients responding or not to the treatment, missing the whole potential of the multivariate information available. Although theoretical properties of the solution could be clearly established, trivial classes are completely useless in real applications, and something more has to be explored. Facing such a complicated phenomenon as rehabilitation, concerned with a lack of clear patterns and difficulties for establishing relationships between exercises and patient improvements, requires indeed to take into account as much prior expert knowledge as possible, even if it is a partial description of the phenomenon. Actually, rehabilitation can be considered an *ISD* and classical hierarchical clustering turns to be unable to capture the complex structure of *ISD* by itself [5].

The additional knowledge supplied by the experts for *ECIBR* regards to variables different than the ones actively considered for the clustering phase of *ECIBR*, and this motivated the generalization of the classical *CIBR* method. In *ECIBR*, the additional knowledge provided by experts is expressed by means of logical rules; it use to be a *partial* description of the domain (as usual for *ISD*, it is very difficult to make explicit a complete *KB* for the domain, and this is a great handicap for using pure AI methods). Here, the knowledge provided by the experts concerned some associations between some sets of tests. For example, they expressed that patients who could not perform the *Sustained Attention Test* neither before nor after the treatment as well as the *Stroop test* will probably have severe impairment; or that patients which can perform more complex tests before the treatment are more likely to improve their condition. The *KB* expressed 6 rules with antecedents involving no more than 6 items of the whole set of 38 variables available. None of the classical statistical methods supports expert

knowledge influencing the analysis. *ECIBR* is a hybrid technique which sensibly improved results by integrating clinical knowledge inside the analysis [9].

Finally, a set of 5 classes was recommended by the system. Several tools were used to assist the interpretation of final classes. From the medical point of view, the composition of the classes is well corresponding with different patterns of response to rehabilitation treatments. All the patients that initially could be assessed appear in a single group. Patients with severe impairment are subdivided in four different profiles which clearly distinguish different response patterns. Among them, the **Dysexecutive group** is identifying an intermediate pattern of partial response to treatment which is qualitatively different from the other groups and merits further research. The **Global Improvement group** is the one with higher response to the treatment showing improvements even greater than the **Assessable group**. From the results of *ECIBR*, experts moved forward onto the analysis of secondary variables among all of them, as prove of principle of further studies with much more available data. They discussed the resulting groups considering the age at TBI and the injury severity:

- **Global Improvement group** contains patients with ages at TBI and injury severity similar to **Resistant group**. The different response to the treatment could be a consequence of clinical variables as injury localization, injury size or their characteristics (contusion, haemorrhage...), not available in this study. In future studies more clinical variables should be considered to identify decisive factors in the patient's prognosis.
- The differences between **Global Improvement** and **Dysexecutive group** could be related to age at TBI. The **Global Improvement group** contains younger patients. Age has been one of the most important prognosis factors in evaluating prognosis after TBI, either at motor and cognitive levels. Thus, in front of a similar initial profile, those patients that suffered a TBI at earlier age would show a more favourable response to the rehabilitation (excepting paediatric population; the influence of age in recovery can be considered from the 16 or 17 years on).
- The interaction between age at TBI and injury severity is also interesting. In general terms, patients of **Global Improvement group** were younger and with lower injury severity than patients of **Dysexecutive group**. Even considering similar cognitive profiles at the beginning of the rehabilitation treatment, injury severity together with patient's age at TBI, could be considered as a predictor of outcome.

ECIBR provided 5 profiles associated with increasing degree of response to the rehabilitation. Meaningful classes were obtained and, from a semantics point of view, the results were sensibly improved regarding classical hierarchical clustering, according to our opinion that *hybrid* techniques that combine AI and Statistics are more powerful for *KDD* than pure ones. *ECIBR* allowed the experts to formulate further hypothesis on their domain to be confirmed in future works and to contribute to increase the knowledge about the different responses of patients with TBI to the rehabilitation.

Acknowledgements

Thanks to the Subsecretaría de Estado de Servicios Sociales, Familia y Discapacidad del Ministerio de Trabajo y Asuntos Sociales; al Instituto de Salud Carlos III y a la Agència d'Avaluació de Tecnologies i Recerca Mèdiques de la Generalitat de Catalunya.

References

- [1] Clohan D.B., Durkin E.M., Hammel J., Murray P., Whyte J., Dijkers M., et al. Postacute rehabilitation research and policy recommendations. *Arch Phys Med Rehabil* 2007;88(11):1535-41.
- [2] Whyte J. Using treatment theories to refine the designs of brain injury rehabilitation treatment effectiveness studies. *J Head Trauma Rehabil*. 2006;21(2):99-106.
- [3] Gibert K., Sonicki Z. Classification based on rules and medical research. *JASMDA* 1999;15(3):319-24.
- [4] Gibert K., et al. Combining a KB system with a clustering method for an inductive construction of models. *LNCS* 1994;89:351-60.
- [5] Gibert K., et al. Clustering based on rules and Knowledge Discovery in ill-structured domains, *Computación y sistemas, revista iberoamericana de computación* 1998;1(4):213-27.
- [6] Gibert K., et al. Knowledge Discovery with clustering: impact of metrics and reporting phase by sing KCLASS. *Neural Network World* 2005;15(4):319-26.
- [7] Lezak MD. et al. *Neuropsychological Assessment*. 4th edition. New York. Oxford University Press, 2004.
- [8] Tukey J.W. *Exploratory Data Analysis*. London. Addison-Wesley, 1977.
- [9] Annicchiarico R., et al. Qualitative profiles of disability. *J Rehabil Res Dev* 2004;41(6A):835-45.
- [10] Gibert K. et al. Impact of Data encoding and Thyroids...., *Technology and Informatics* 2001;90:494-503.
- [11] Comas J. et al. Knowledge Discov... by means of inductive methods in WWTP. *AICom* 2001;14(1):45-62
- [12] Rodas J. et al. KDSM Methodology for Knowledge Discovery from Ill-Structured Domains presenting very short and repeated serial measures with blocking factor, *LNAI* 2001;2504:228-38.
- [13] Salvador L. et al. Study DEFDEP: operational definition of dependency in psychic disabled persons. Technical report to the Spanish Dependency Law delivered to Generalitat de Catalunya. Jul 2007.
- [14] Brian Everitt, "Cluster Analysis," Second Edition, Social Science Research Council, Heinemann Educational Books, London, Halsted Press, Division of John Wiley & Sons, NY

Using Ensemble-Based Reasoning to Help Experts in Melanoma Diagnosis

Ruben NICOLAS ^a, Elisabet GOLOBARDES ^a, Albert FORNELLS ^a,
Sonia SEGURA ^b, Susana PUIG ^c, Cristina CARRERA ^c, Joseph PALOU ^c, and
Josep MALVEHY ^c

^a *Grup de Recerca en Sistemes Intel·ligents
Enginyeria i Arquitectura La Salle - Universitat Ramon Llull
Quatre Camins 2, 08022 Barcelona
{rnicolas,elisabet,afornells}@salle.url.edu*

^b *Dermatology Department of Hospital del Mar
Passeig Marítim 25-29, 08003 Barcelona
ssegura@imas.imim.es*

^c *Melanoma Unit, Dermatology Department
Hospital Clinic i Provincial de Barcelona, IDIBAPS
{spuig,ccarrera,jpalou,jmalvehy}@clinic.ub.es*

Abstract. New habits in solar exposure have caused an important increase of melanoma cancer during the last few years. However, recent studies demonstrate that early diagnosis drastically improves the treatment of this illness. This work presents a platform called MEDIBE that helps experts to diagnose melanoma. MEDIBE is an ensemble-based reasoning system that uses two of the most important non-invasive image techniques: Reflectance Confocal Microscopy and Dermatoscopy. The combination of both image source improves the reliability of diagnosis.

Keywords. Artificial Intelligence in Medical Applications, Case-Based Reasoning, Ensemble Methods, Melanoma, Basal Cell Carcinoma, Reflectance Confocal Microscopy, Dermatoscopy.

Introduction

Death related to melanoma cancer has increased during the last few years due to the new solar habits. This growth convert it in a very important case study in our society, even more if we analyze that this kind of cancer affects people of any age, but especially young ones. According to the American Cancer Society, it is the skin cancer which causes most deaths because it is mortal in approximately twenty percent of cases [11,21]. Although an early diagnosis allows practically a secure regain, its identification is not trivial due to the different sizes, shapes and colors in which it can appear [16].

The Dermatology Department of *Hospital Clinic i Provincial de Barcelona* (HCPB) works with two of the most promising techniques of image analysis for melanoma di-

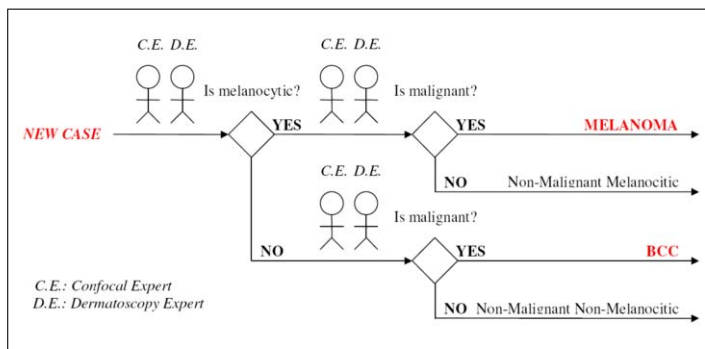


Figure 1. Medical protocol followed by experts from HCPB for melanoma diagnosis.

agnosis: Dermatoscopy and Reflectance Confocal Microscopy (RCM) [19]. The first is based in the microscopical image created with epiluminiscence (x10.30) and the second makes the image with the reflectance of a coherent laser (x100) with a cell resolution [18]. Through the application of both analysis they perform a diagnostic process based on two steps. First, they analyze if the new case is melanocytic or not. Afterwards, they assess if this case is malignant or not. Thus, the combination of both diagnosis allows experts to determine if the new case is Melanoma, Basal Cell Carcinoma (BCC) or a non-malignant tumor as figure 1 shows. Moreover, one benefit of this approach is that expert does not need to perform any invasive test to the patient.

The goal of this work is to develop a framework to help experts to automate the diagnosis process under their supervision. The approach models the protocol described in figure 1 through several ensemble systems which contain the logical decision performed by Confocal and Dermoscopic experts. The ensemble is based in Case-Based Reasoning [1] because the approach uses past experiences to solve new cases and this is exactly the same procedure used by experts.

The paper is organized as follows. Section 1 describes some related work. Section 2 describes the medical application. Section 3 analyzes its performance. Section 4 ends with conclusions and further work.

1. Related Work

Ensemble methods combine the decisions from different systems to build a more reliable solution using the individual ones [12,13,15,20]. The combination of approaches can be summarized in: 1) Bagging, 2) Boosting, and 3) Stacking. Bagging [4] and Boosting [7,6] are based in the combination of the outputs using votes. In concrete Bagging replicates N systems of the same approach but using different data sources. In opposition Boosting follows the same idea but it defines models in order to complement them. On the other hand Stacking [5] is based on heuristics that combine the outputs of several approaches. As voting methods the most common ones [14] are: 1) Plurality, 2) Contra-Plurality, 3) Borda-Count, and 4) Plurality with Delete. All these methods are based on the number of votes of a class (plurality) but with multiple types of addition of plurality and decision of better class.

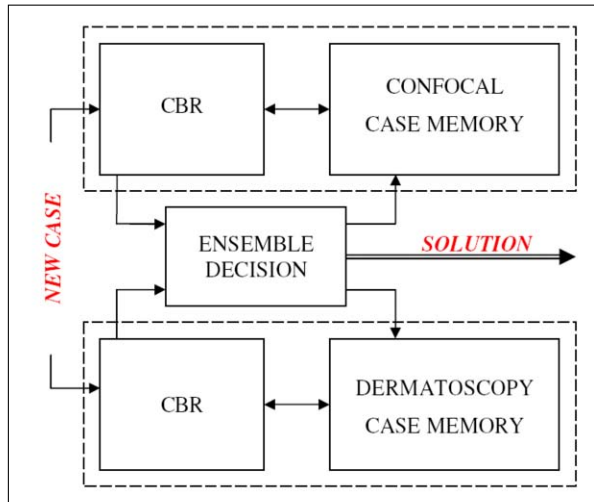


Figure 2. Ensemble Decision Schema.

Although there are works focused on studying the melanoma domain from individual approaches such as in [9], the application of ensemble methods has increased in last years. One of this lines is to improve clustering using ensembles [3]. There are also works to allow the classification using data of different complexity [2] and with different types of medical information [8,10]. In contrast to these approaches we would like to classify in melanoma domain following the medical diagnosis protocol using different ensemble classifiers. Attending to this problem, the medical necessities and the existing data seems interesting to create an ensemble model with an expert for each kind of data. We note that we adapt the model using different attributes of the same data in each ensemble, then the independence of the data is guaranteed, in contrast to the standard Bagging. Analysing that the classification attributes are boolean the vote method should be based on plurality but with some arranges requested by medical researchers, who weight more the information of an specific machine (Confocal Microscopy).

2. MEDIBE: A framework for melanoma diagnosis

MEDIBE (MElanoma DIagnosis Based on Ensembles) is a computer aided system for melanoma diagnosis based on the medical protocol described in the first section. For each one of the decision points, an ensemble system is used to answer the medical question using the knowledge extracted from the Dermatoscopy and the Reflectance Confocal Microscopy image data as Fig. 2 shows.

As we can observe, the ensemble system combines the output of two Case-Based Reasoning (CBR) systems [1]. This is because CBR performs the same resolution procedure than experts: solving new cases through the comparison with previously solved cases. In a general way, the CBR life cycle can be summarized in the next four steps: 1) Retrieving the most similar cases from the case memory with the assistance of a similarity function; 2) Adapting the retrieved solutions to build a new solution for the new case; 3) Revising the proposed solution, and; 4) Retaining the useful knowledge gener-

```

Let  $c_{new}$  be the new input case
Let  $best_{confocal}$  be the most similar case using the confocal CBR
Let  $best_{dermatoscopical}$  be the most similar case using the dermatoscopical CBR
Let  $distance(c_i, c_j)$  be the distance between two cases  $c_i$  and  $c_j$  performed by the
normalized Euclidean distance
Let  $threshold_{confocal}$  be the minimal value to accept two cases as similar from the
confocal point of view
Let  $threshold_{dermatoscopical}$  be the minimal value to accept two cases as similar from
the dermatoscopical point of view
Let  $class(c)$  be the class of the case  $c$ 
if  $distance(c_{new}, best_{confocal}) < threshold_{confocal}$  then
  return  $class(best_{confocal})$ 
else
  if  $distance(c_{new}, best_{dermatoscopical}) < threshold_{dermatoscopical}$  then
    return  $class(best_{dermatoscopical})$ 
  return  $class(best_{confocal})$ 

```

Figure 3. Algorithm to diagnose a new case using the confocal and dermatoscopical criteria.

ated in the solving process if it is necessary. Thus, the explanation capability is highly appreciated by the experts because they are able to understand how the decisions are done. Each one of the CBR systems feed from two different case memories which stores all the previously diagnosed injuries through the confocal and the dermatoscopy studies respectively. These two parts are completely independent and at the end of its work they put on its vote for the best classification according to their specific data. With this separate ballots the system creates the final diagnosis (Solution) and, if proceed, save the new case in one of the case memories or both.

The decision process followed to perform a diagnosis is described in figure 3 and it represents the logical used by the experts. In spite of using a collaborative scheme where both diagnosis are combined, experts mainly focus on confocal diagnosis and, only if the diagnosis is non conclusive they use the dermatological diagnosis. Therefore, the selection of the threshold values used to decide if the relevance diagnosis is the confocal or the dermatological are crucial to achieve a good performance. Both values need to be defined by experts.

3. Experimentation

This section describes the data extracted from images and analyzes the results of the experiments performed with MEDIBE through sensitivity and specificity rates.

3.1. Testbed

The classification of injuries in melanoma domain is not trivial. One of the main difficulties is the huge amount of information that new technologies are able to collect, and the ignorance about how they are related. One of the most used techniques to gather information from tissue is the dermoscopic analysis. Nevertheless, there are specific kinds

Table 1. Classification accuracy of MEDIBE using only confocal images, only dermoscopic images, and both images.

| | Melanocytic | Melanoma | BCC |
|--------------------------------|-------------|----------|-----|
| Confocal Image | 89% | 87% | 96% |
| Dermatoscopy Image | 88% | 79% | 90% |
| Confocal and Dermoscopic Image | 95% | 89% | 97% |

Table 2. T-test comparison between methods using 95% of confidence level

| | Melanocytic | Melanoma | BCC |
|-------------------------|-------------|----------|-----|
| Ensemble - Confocal | ↑ | - | ↑ |
| Ensemble - Dermatoscopy | ↑ | ↑ | ↑ |
| Confocal - Dermatoscopy | - | ↑ | ↑ |

of melanoma that it is not able to diagnose [17]. For this reason is why the experts want to evaluate if a new technique, called confocal analysis, is able to detect them or if the usage of both techniques can improve the individual analysis.

The dataset used in this work is composed by 150 instances of suspicious lesions from HCPB’s patients. All instances contain information related to Confocal Image, Dermatoscopy Image and Diagnosis (corroborated with the histology). Attending the considerations of the medical experts that have created this set, it includes enough cases from each kind of illness (or healthiness) to be representative of the domain. Then, in medical terms it is an appropriated case memory for this study. Detailing the instances, dermatological and confocal image data are collected from two different microscopes with different kinds of precision. The first one, known as Dermatoscopy, have forty-one fields with knowledge of Symmetry, Color, Reticle, Globules, and other concrete aspects [17]. The second one, called Reflectance Confocal Microscopy (RCM), have data from eighty-three different attributes with information about Vessels, Nucleated and Non-Nucleated Cells, Dermal Papilla, Pagetoid Infiltrations, Basal Cell, and other [18]. RCM contributes with more attributes due to its higher resolution than Dermatoscopy.

Figure 4 contains images of four different types of injuries: A) Melanoma in situ, B) Melanoma, C) Nevus (melanocytic non-melanoma), and D) Basal Cell Carcinoma. For each one of types, different level of details are available: 1) Clinical image, 2) Dermatoscope image, 3) In vivo Confocal image, and 4) Histo-pathological study with hematoxylin eosine stain. It is important to highlight that the different types of image information could give very different aspects even in same final diagnostic cases. This situation is clear comparing A and B part where we have two melanomas but in first case A the lesion studied is clinically and dermoscopically banal (non-suspicious of malignancy) but confocal is characteristic of melanoma as is demonstrated in the histology. In case B the diagnosis of melanoma is clear in all images. Part C and D permits to see that in non-melanoma melanocytic cases and in BCC ones clinical images are similar but under dermatoscopy and confocal microscopy have specific criteria to reach the correct diagnosis, nevus (benign melanocytic lesion) and BCC (malignant non-melanocytic).

3.2. Experimentation, Results and Discussion

We have tested the classification accuracy of MEDIBE and the one of independent CBR systems (one for confocal data and another for dermatoscopy). Both experimentations

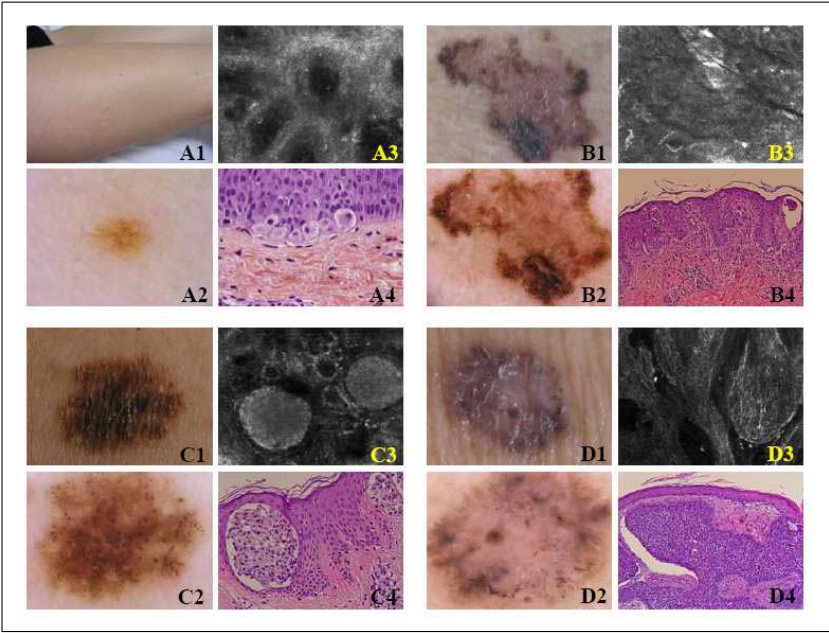


Figure 4. Images from cases included in the study.

Table 3. Sensitivity and specificity results from MEDIBE using combined information of confocal and dermoscopic images.

| | Melanocytic | Melanoma | BCC |
|-------------|-------------|----------|------|
| Specificity | 91% | 96% | 100% |
| Sensitivity | 96% | 66% | 88% |

uses a CBR configured with one-nearest neighbor algorithm with normalized Euclidean distance as retrieve function. We have tested different confocal and dermatoscopic thresholds according to medical experts criteria. Finally we have reached a consensus to use 0.5 as confocal threshold and double of the distance between new case and best confocal case as dermatological one. With this experiment framework we have applied a 10-fold stratified cross validation to the original data to obtain the average accuracy of those systems.

Analyzing these results, table 1 shows the accuracy rate to classify new injuries in the three possible classes (Melanocytic, Melanoma and BCC) using only confocal image, only dermoscopic image and both images. To evaluate the representativity of accuracy differences table 2 shows the results obtained from the application of t-test (at 95% of confidence level) to the accuracy results of different data configurations. We represent with the symbol (↑) if the first method is significantly better than the second one, and with the symbol (-) if its significance is not representative. Results of accuracy and significance, highlight two points: First, confocal analysis diagnoses better than dermatoscopy (with significant difference in all cases except in melanocytic differentiation where are equivalent). Second, the combination of both data improve the results (despite of the case of Melanoma classification where the use of only confocal information is equivalent to the combined one).

On the other hand, table 3 summarizes the results of analysing the statistics from the point of view of specificity and sensitivity. They show that is more reliable to do a prognostic of real negative cases than the positive ones. This fact is because datasets are unbalanced, that is, they have different number of cases of each type because datasets represent a real situation: there are more healthy than sick people.

Detailing negative cases, we could say that in melanocytic distinguish the percentage of bad classification is nine percent, in melanoma four percent, and at last in BCC the system does not fail in any case. Notwithstanding positive cases classification give worse results, the sensitivity of classification are better than the obtained without the application of the proposed method. As *e.g.*, doing the analysis of the worst sensitivity result of the system (66% obtained in melanoma classification) it is better than the 57% that have medical researchers in difficult melanoma classification cases nowadays.

4. Conclusions and Further Work

Melanoma diagnosis using non-invasive techniques is nowadays one of the most important goals in dermatology, due to the necessity of getting an early diagnosis. The diagnostic process is complex because of the highly variability of shapes, sizes, and colors.

We propose a platform called MEDIBE for automatizing medical protocol followed by HCPB's experts to diagnose a melanoma. The application combines information from two of the most promising techniques based on images through an ensemble algorithm founded on experts' experiences. After the analysis of results from testing melanoma dataset, we can conclude that the combination of both images improves the individual results applying the medical protocol.

The further work is focused on two lines. First, create a new specific voting method not conditioned by medical criteria. This independence with experts could permit to find aspects not included in formal algorithms but used unconsciously by them, as clinical details. Second, it would be useful to test a different ensemble method idea in diagnostic. It is to create a system that permits to prognostic if a lesion is malignant or not with independence to its concrete type. Then, the different ensembles could be trained to vote according to specific problem (melanocytic, melanoma, BCC, and other) and not in reference to one type of image. Finally, this ensemble has to vote if the new case needs treatment or not.

Acknowledgements

We thank the Spanish Government for the support in MID-CBR project under grant TIN2006-15140-C03 and the Generalitat de Catalunya for the support under grants 2005SGR-302 and 2008FI_B 00499. We thank Enginyeria i Arquitectura La Salle of Ramon Llull University for the support to our research group. The work performed by S. Puig and J. Malveyh is partially supported by: Fondo de Investigaciones Sanitarias (FIS), grant 0019/03 and 06/0265; Network of Excellence, 018702 GenoMel from the CE.

References

- [1] A. Aamodt and E. Plaza, Case-Based Reasoning: Foundations Issues, Methodological Variations, and System Approaches, *AI Communications* **7** (1994), 39–59.
- [2] R.E. Abdel-Aal, Abductive network committees for improved classification of medical data, *Methods of Information in Medicine* **43** (2004), 192–201.
- [3] R. Avogadri and G. Valentini, Fuzzy ensemble clustering for DNA microarray data analysis, *LNCS 4578 LNAI* (2007), 537–543.
- [4] E. Bauer and R. Kohavi, An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants, *Machine Learning Journal* **36** (1999), 105–139.
- [5] B. Clarke, Comparing Bayes model averaging and stacking when model approximation error cannot be ignored, *J. Mach. Learn. Res.* **4** (2003), 683–712.
- [6] Y.S. Dong and K.S. Han, Boosting SVM classifiers by ensemble, *WWW '05: Special interest tracks and posters of the 14th International Conference on World Wide Web* (2005), 1072–1073.
- [7] G. Eibl and K.P. Pfeiffer, Multiclass Boosting for Weak Classifiers, *J. Mach. Learn. Res.* **6** (2005), 189–210.
- [8] A. Fornells, E. Golobardes, E. Bernadó and J. Martí, Decision Support System for Breast Cancer Diagnosis by a Meta-Learning Approach based on Grammar Evolution. *9th International Conference on Enterprise Information Systems* (2006), 222–227.
- [9] A. Fornells, E. Armengol, E. Golobardes, S. Puig and J. Malveyh, Experiences Using Clustering and Generalizations for Knowledge Discovery in Melanomas Domain. *7th Industrial Conference on Data Mining* **5077** (2008), 57–71.
- [10] Y. Gao et al., LCSE: Learning classifier system ensemble for incremental medical instances, *LNCS 4399 LNAI* (2007), 93–103.
- [11] R.M. Gutierrez and N. Cortes, Confronting melanoma in the 21st century, *Med Cutan Iber Lat Am* **35** (2007), 3–13.
- [12] D.A. Hull, J.O. Pedersen and H. Schütze, Method combination for document filtering, *SIGIR '96: Proceedings of the 19th Annual International ACM SIGIR Conference on Research and development in information retrieval* (1996), 279–287.
- [13] N. Indurkha and S.M. Weiss, Solving regression problems with rule-based ensemble classifiers, *KDD '01: Proceedings of the seventh ACM SIGKDD International Conference on Knowledge discovery and data mining* (2001), 287–292.
- [14] K.T. Leung and D.S. Parker, Empirical comparisons of various voting methods in bagging, *KDD '03: Proceedings of the ninth ACM SIGKDD International Conference on Knowledge discovery and data mining* (2003), 595–600.
- [15] P. Melville and R.J. Mooney, Diverse ensembles for active learning, *ICML '04: Twenty-first international conference on Machine learning* (2004).
- [16] R. Nicolas, E. Golobardes, A. Fornells, S. Puig, C. Carrera, and J. Malveyh, Identification of Relevant Knowledge for Characterizing the Melanoma Domain. In *2nd International Workshop on Practical Applications of Computational Biology and Bioinformatics* (2008), In Press.
- [17] S. Puig et al., Melanomas that failed dermoscopic detection: a combined clinicodermoscopic approach for not missing melanoma, *Dermatol Surg* **33** (2007), 1262–1273.
- [18] A. Scope et al., In vivo reflectance confocal microscopy imaging of melanocytic skin lesions: Consensus terminology glossary and illustrative images, *J Am Acad Dermatol* **57** (2007), 644–658.
- [19] S. Segura et al., Dendritic cells in pigmented Basal cell carcinoma: a relevant finding by reflectance-mode confocal microscopy, *Arch Dermatol* **143** (2007), 883–889.
- [20] W.N. Street and Y.S. Kim, A streaming ensemble algorithm (SEA) for large-scale classification, *KDD '01: Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining* (2001), 377–382.
- [21] M.A. Tucker and A.M. Goldstein, Melanoma etiology: where are we?, *Oncogene* **22** (2003), 3042–3052.

Ergonomic Advice through Case-Based Reasoning to Avoid Dangerous Positions Adopted Using the Computer

Fernando Orduña Cabrera (KELMG)^{a,c} Miquel Sànchez-Marrè (KELMG)^{a,1}
 Jesús Miguel García Gorrostieta (RIIIA)^b Samuel González López (RIIIA)^{c,2}

^a *Technical University of Catalonia (UPC)*

Campus Nord-Building Omega Software Dept. (LSI)

Jordi Girona 1-3, E08034, Barcelona, Spain

{forduna, miquel}@lsi.upc.edu

^b *Universidad de la Sierra*

División de Ingenierías y Tecnologías

Moctezuma, Sonora, México

miguel.garcia@universidaddelasierra.edu.mx

^c *Instituto Tecnológico Superior de Cajeme*

Division de Investigación y Posgrado

Cd. Obregon, Sonora, México

samuelgonzalezlopez@gmail.com

Abstract. Nowadays a computer is used as a main tool for many tasks. Usually a session using a computer in a computer center takes from 50 up to 150 minutes, getting the user to be tired. With this research, the dangerous positions adopted by the students in a computer center are analyzed and recommendations of exercises are listed. The exercises avoid the pain feel caused by those hazard positions. The suggestions are notified to the user by the Ergonomic-CBR application developed. After 40 minutes of a session with the computer, the Ergonomic-CBR application sends a test to user. With the answers, a case is formulated and the CBR cycle is loaded. The similar case is retrieved and adapted using expert-coded rules. All adaptations are done according to the pain feel. The result of the experiments done in Universidad de la Sierra, reflect that the students suffer of repetitive tension related with using computers. The experimental groups presented a diminution in stress and a huge acceptance of relaxation exercises proposed by the Ergonomic-CBR system.

Keywords. Case-Based Reasoning, Ergonomics

1. Introduction

The extensive use of computers is associated with different types of pathologies. In the report [6], some diseases related with this task are outlined, and specially the diseases

¹KELMG (Knowledge Engineering and Machine Learning Group)

²RIIIA (Red Iberoamericana de Investigación en Inteligencia Artificial)

in superior members are mentioned. The conditions normally found are indicated in the investigations of [2], and include the syndrome of the carpal tunnel, backache, muscular tension, syndrome of tension in the neck and tendonitis. The prolonged use of computers and bad body positions are frequently the cause to have health problems, mostly related to the eyes and the muscle-skeletal system. The study [1] depicted that a 61.5% of people directly working with computer monitors, have pain in the neck and shoulders which can interfere with their work.

The research work of [16] depicts an instrument that validate the adequation of the design of the workstation based on the requirements of the worker, and evaluate the incorrect position in front of the computer screen. They summarize in one table some possible implications of work elements in the position adopted. They show some recommendation for the workstation design.

The investigation [10], which was applied to 206 students in the University of California, where most of the students use the computer for about 40 hours a week, the 60% of students feel pain in the neck when they use the computer. The most frequently pains were in the fingers, hands and wrists; in second place, the shoulders and neck; and finally, the forearm and the elbow. The level of pain found was of 4.5 (scale of the 0-10, being 0=none and 10=unbearable).

The studies [5,1,9,6] make the following recommendations when working with computer:

1. Adjust the screen slightly below the level of the eye, with an inclination of 10 to 20 degrees and 50 or 65 cm away from the eyes.
2. Each 40 - 45 minutes of work take a rest from 5 to 10 minutes, focus the look on objects that are at more than 6 meters distance.
3. Blink regularly to avoid the dryness of the eyes.
4. Use lens with ultraviolet protection and antireflection.
5. Visit the doctor for an examination of the eyes once a year.
6. Use ergonomic keyboards.

In [3], different durations of work sessions from 300850 users in the United Kingdom were analyzed. The result of this study presents that taking a rest every 45 minutes presents the best results to avoid any kind of pain related to the computer use.

The aim of this paper is to develop a CBR application to find and adapt the most suitable list of exercises like advice to reduce some of the symptoms named before, specially focused on the arm annoyances. The advices are generated by a Case-Based Reasoning application (Ergonomic-CBR). The Ergonomic-CBR tool starts when a computer session is started. After 40 minutes a set of questions is loaded. These questions must be answered by the computer user. With the answers, the tool build a case and retrieve the most similar case, then adapt the retrieved case according to the new situation. The initial cases have a list of exercises. This list has a direct relation between symptom (the questions) and exercises to reduce the pain feel.

The exercises must be adapted according to the level of pain feel, which is know by the user answers. For example, if a computer user has high annoyances, the exercises recommended must be done several times. To present a correct recommendation, the case retrieved is adapted to the new situation and the exercises are adapted using rules too. Another relevant adaptation pattern is when the exercises are incomplete and an extra exercise is required to calm the pain feel.

The development of the Ergonomic-CBR was done using Caspian CBR Shell tool and the experiment was tested in a computer center of the Universidad de la Sierra with 5 groups of students. When the test was finished, the people who did the exercises got a better feel than the people that did not do the exercises. The exercises were well accepted by the students.

2. Architecture of the ergonomic system

The figure 1 depicts the dynamics of the Ergonomic-CBR system. The first part (the gray tasks) represents the interaction with the user, starting with an identification of the user and then the user attend the class. After 40 minutes the system ask questions evaluating the physical health. Once the answers have been obtained the CBR part is started. The answers constitute the new case to be solved by the Ergonomic-CBR system. After the CBR cycle is completed, a list of exercises to be done is explained to the user. Next, the temporal cycle starts again.

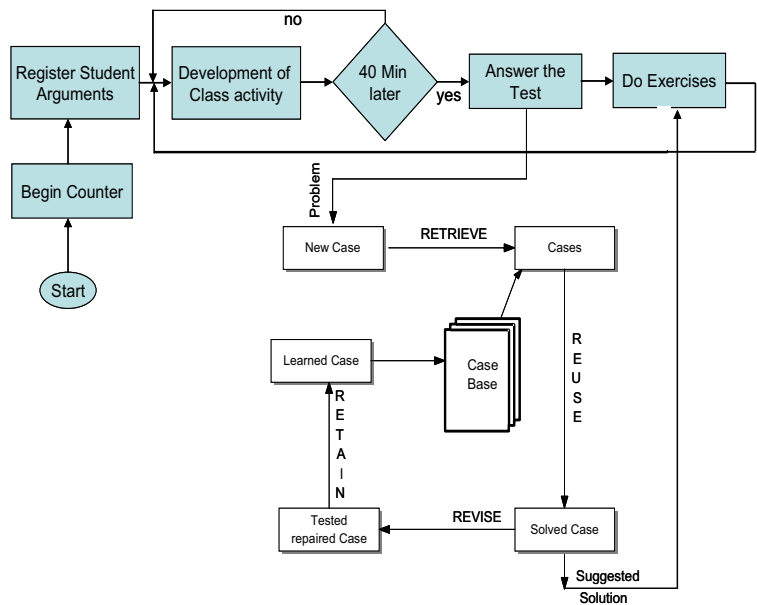


Figure 1. Architecture of the system.

3. The Case Base and the retrieval process

Ergonomic-CBR has an initial case base of 64 previous designs cases. The cases are most representative that could be presented in the system. The cases were designed by experts, on considering a possible combination of the six evaluation questions. As result of this combination, the sixty four cases were developed. In the design of the cases, research

works such as [4,3,7] have been considered by the experts, and used as a reference to build the case attribute structure. The six attributes used to characterize the cases are the following:

1. Do you feel pickets in the hands? *None, Low, Medium, High*
2. Do you feel heat in the basis of the hand? *None, Low, Medium, High*
3. Do you feel pain in the basis of the hand? *None, Low, Medium, High*
4. Do you feel pain in the elbow? *None, Low, Medium, High*
5. Do you feel weakness in any area of the arm? *None, Low, Medium, High*
6. Do you feel pain in wrists? *None, Low, Medium, High*

The solution of a case is formed of a list of corresponding exercises. The list of exercises for each case was assigned taking the research works in [5,6]. The list is:

1. Place the hands interlaced to the front for 10 seconds
2. Raise the arms up and close and open your hands 5 times. (Repeat 2 times)
3. Turns your wrists 5 times in clock way and 5 times in the opposite way.
4. Try to always rest your arms on the table.
5. Visits the doctor for a checkup.

3.1. A Conceptual Ergonomic-Case is presented as:

For instance, a case could be the follow one:

Do you feel pickets in the hands? Answer **Low**

Do you feel heat in the basis of the hand? Answer **High**

Do you feel pain in the basis of the hand? Answer **Medium**

Do you feel pain in the elbow? Answer **None**

Do you feel weakness in any area of the arm? Answer **Medium**

Do you feel pain in wrists? Answer **Low**

The exercises needed to solve the pain feel in this case are:

1. Place the hands interlaced to the front for 40 seconds
2. Raise the arms up and close and open your hands 5 times. (Repeat 6 times)
3. Turns your wrists 5 times in clock way and 5 times in the opposite way (Repeat 2 times).
4. Try to always rest your arms on the table.
5. Visits the doctor for a checkup.

The retrieval process is a sequential process of the complete list over the case base. The similarity process is under taken, using Index Definitions, generating an index structure to improve the search. Indexing on an appropriate field helps to ensure that the retrieved cases are in the right ball-park. Indexes are intended to prune the matching process so that many cases are rejected early and a lot of work is saved.

4. Adaptation of the exercises to be done

The list of exercises recommended by the system was elaborated taking as a basis the works in [5,6]. The list is:

1. Place the hands interlaced to the front for 10 seconds
2. Raise the arms up and close and open your hands 5 times. (Repeat 2 times)
3. Turns your wrists 5 times in clock way and 5 times in the opposite way.
4. Try to always rest your arms on the table.
5. Visits the doctor for a checkup.

Each exercise is related to relax one or more symptoms. The relation is depicted in the figure 2.

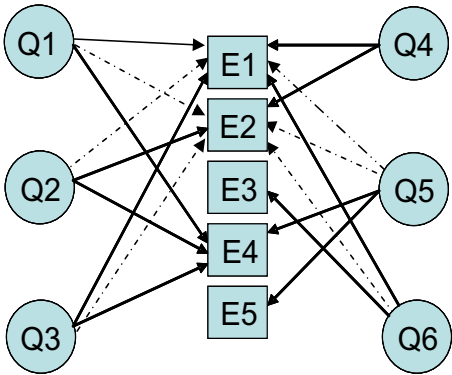


Figure 2. Relation between Tasks and Symptoms.

The figure 2 has circles representing the questions (case attributes) and boxes representing exercises (case solutions). The direct relation between questions and exercises are depicted by the continue arrows, and dots arrows mean an indirect relation. An indirect relation is presented in special cases. For instance suppose the user only has answered the Question One (Q1) and Question Three (Q3). The exercises corresponding to Q1 are: Exercise One (E1) and Exercise Four (E4). And for Q3 the exercises related are: E1 and E4. So, for Q1 and Q3 the exercises are E1 and E4, but Q1 and Q3 have an indirect relation with E2. As the symptoms are represented by the answers of Q1 and Q3, the indirect exercise E2 must be added to the list, getting the final list: E1, E2 and E4.

4.1. Intensity of the exercise to be done

The number of repetition of the exercise required to solve the pain is based on the intensity of the pain feel. If the symptom is high, the exercise must adapt to solve a high pain.

In [14], presents a research focused on the hand and wrist pain. Quantification of pain in both observations for experimentation is complex; the algometric analysis is based on that pain, By itself is difficult considering so many parameters to define. In [14], it is depict a verbal validation scale using words to evaluate the intensity of pain feel or the pain relief, and describe the actual categories used nowadays, those categories are the used on our test to evaluate the pain feel (none, low, medium, high).

Is understanding that the incapacity of the problem cause in hands pain, fosters frustration, anxiety and eventually alteration in family and social dynamic according to the intensity of the disease if that is the case. A detail of the levels evaluated is depicted as:

- None: is presented when the student hasn't any pain.
- Low: is considered low when the student starts to feel a tiny pain.
- Medium: it's a constant pain but can deal with it and continue working.
- High: it's a strong and constant pain, and can't work.

To do this adaptation between pain feel and intensity level of exercise consider the next CASL rule.

```
REPAIR RULE AdapHighLevel IS
WHEN  wrist_pain IS High
THEN  EVALUATE Repetitions TO Repetitions * 1.75;
END;
```

Where the Repetitions represent the intensity of the exercise, which in this case means an of 75% in exercise times.

5. How the ergonomic the CBR works

The Caspian CBR Shell [11] has been used to develop the Ergonomic-CBR application. Caspian is a case-based reasoning system capable of dealing with realistic case-based reasoning problems, developed by the University of Wales. And CASL is a language used for CBR. The contents of a case-base are described in a file know as a case file using the language CASL. The program Caspian uses this case file to create a case-base in the computer's memory. The structure of the case file is guided by the next syntax structure:

[Introduction]: Comments for an introduction to the CBR system.

Case Definition: Defines the types and the weights of the problem fields that may appear in a case.

Index Definition: Defines the fields used as indexes when searching for a matching case.

[Modification Definition]: Defines the modification rules.

[Pre-processing Rule Definition]: Defines rules which are used to alter a case.

[Repair Rule Definition]: Contain the repairs rules.

Case Instance {Case Instance}: These are the cases which make up the case base.

5.1. Format of a case in Caspian Shell

For a case, consider that Y and Z could take the values:

Y = [wrist_pain or weakness_of_arm or annoyances_in_elbow or hand_basis_weakness or hand_basis_heat or pickets_in_hand]

Z = [Non or Low or Medium or High]

The format of a case for Ergonomic-CBR is:

```
CASE INSTANCE case_Symptom_X IS
    wrist_pain = Z;
```

```

        weakness_of_arm = Z;
        annoyances_in_elbow = Z;
        hand_basis_pain = Z;
        hand_basis_heat = Z;
        pickets_in_hand = Z;
SOLUTION IS
        Do_exercise = [list of exercise];
        LOCAL REPAIR RULES (See adaptation
                           section for details)
END;
```

5.2. Adaptation from the best matching case

Each one of the 64 cases has a particular adaptation solution based on local rules, which defines the solution for a presented case. The adaptation is based on the combination of the 4 possible answers {Z} with 6 possible symptoms {Y}. Then, those combinations are the typed by the user when answer the questions formulated by the system, with these set of answers and question (SetAQ) a new case is formulated, and begin the cycle of the CBR. The local rules implemented in the Ergonomic-CBR for adaptation follow the next format:

```

LOCAL REPAIR RULE DEFINITION IS
    REPAIR RULE AdapSolution_based_on_SetAQ IS
    WHEN
        SetAQ = {Y[1 to 6] = Z || 1 to 6 are the
                symptoms and Z the pain level }
    THEN
        EXTRACT exercise FROM [list of exercises];
        CHANGE Do_exercise TO exercise;
    END;
```

6. The experiment

In section 4 was depicted how the research work [14] (done by specialists of the Ortopedistas Hospital ABC), presents a verbal validation scale where using words to evaluate the intensity of pain feel or the pain relief. Those categories are the used on our test to evaluate the pain feel and are depicted in the same section.

Section 3 presents a set of six questions used in the test applied to the experimental groups, and the figure 1 describe how often the test was applied.

In order to assess the pain reduction level using the Ergonomic-CBR prototype or not using Ergonomic-CBR prototype, the experimental validation was applied to 5 groups in a computer center, with sessions of 100 minutes. A total of 103 students answered the test. The 5 groups were divided in 3 *experimental groups* and 2 *control groups*. The *control groups* answered the test and did not do any relaxation exercises of the list in section 3, and the *experimental group* has did the exercises listed in section 3.

Figure 3(a) depicts that 56% of the students experienced some pain after 40 minutes of work in the computer. This result includes the *experimental* and *control groups*. After

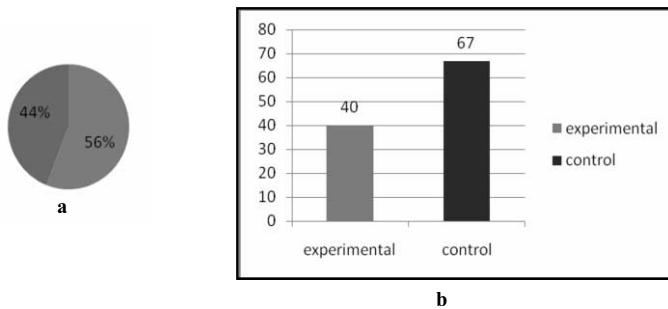


Figure 3. (a) Presented some pain after 40 minutes (b) Feel pain, experimental and control groups

80 minutes of work in front of the computer, the test was applied again to the *control group* in which 67% of the students presented some kind of pain, indicating that there was an increase in the number of people who experienced some pain. However, in the *experimental group* only the 40% of students felt pain, decreasing the number of affected students. The figure 3(b) compares the percentage of students with pain, including the *experimental* and *control groups*.

7. Conclusions and Feature work

From the experimental work done, it can be observed that a 56% of students suffer some kind of pain after 40 minutes of work in front of the computer. After doing the exercises provided by the system only 40% of the students feel any kind of pain. These results indicates that Ergonomic-CBR system could help in a computer center reducing the physical fatigue and pain.

The Ergonomic-CBR could provide a more precise exercises, as it can adapt the duration, intensity and repetition features of each exercise. In fact, it can improve the human recommendation of exercises by means of a more suitable and personalized exercises for each person, specially in the arm such as in this case. A complete Ergonomic-CBR system, avoiding the hazard positions for the muscle-skeletal system, could be built for a deeper ergonomic system.

The Ergonomic-CBR application is a prototype of a complete ergonomic system and could offers to evaluate physical aspects such as ocular annoyances, headache, muscle-skeletal pains produced by inadequate positions when working with the computer. Could be managed a full version of an ergonomic system could evaluate symptoms such as: neck pain, back pain, shoulders pain, arms pain, wrists paint, elbow pain, fingers pain, and some eyes feels such as eyestrain, dry eye, red eye, wet eye, hot eye, blurred vision, bad targeting, Diplopia, and headache. In this research, ours efforts are specially focused on the arms pain, wrists paint and elbow pain. In near feature, others aspects such as those discussed above, will be added to the CBR advice system, improving the current version.

References

- [1] Bongers, P., Kremer, A. y Laak, J. 2002. Are psychosocial factors, risk factors for symptoms and signs of the shoulder, elbow, or hand/wrist?: A review of the epidemiological literature. *American Journal of Industrial Medicine*. 2002; 41(5): 315-342.
- [2] Del Río Martínez, Jesús Heraclio y González Videgaray, MariCarmen. 2007. Trabajo prolongado con computadoras: consecuencias sobre la vista y la fatiga cervical. IX Congreso Internacional de Ergonomía, Abril 26-28, 2007, Sociedad de Ergonomistas de México A.C, D.F. México.
- [3] Heasman, T., Brooks, A. y Stewart T. 2000. Health and safety of portable display screen equipment. Reporte de Investigación de Health & Safety Executive. Londres, Reino Unido.
- [4] Hupert Nathaniel, C. Amick Benjamin, H. Fossel Anne, M. Coley Christopher, M. Robertson Michelle and N. Katz Jeffrey. 2004. Upper extremity musculoskeletal symptoms and functional impairment associated with computer use among college students. IOS Press.
- [5] Pérez Aguilera, María de Jesús y Hernández Arellano, Juan Luis. 2007. Identification of High Risk Conditions on Teachers in the Distance. Primer Congreso Internacional de Ingeniería Industrial, Mecatrónica y Manufactura, Octubre 4 - 6, 2007, Universidad Autónoma de Ciudad Juárez, Ciudad Juárez, México.
- [6] Robertson, Vivienne. 2004. 10 Tips for Comfortable Working [en línea], [citado el 1 de Octubre del 2007]. Disponible en <http://www.system-concepts.com/articles/article0076.html>
- [7] Saadat Sohrab, Zhang Shujun, Gaunt, Trevor y Hapeshi, Kevin. 2006. Are Students Aware of Ergonomics of Computer Uses. World Automation Congress 2006, July 24-26, Budapest, Hungary.
- [8] Williams, Inger M. 2002. Student's Musculoskeletal and Visual Concerns. En XVI Annual International Occupational Ergonomics and safety Conference.
- [9] Quintanilla Kuny, Paulina y Médez Kury, Valeria. 2007. Síndrome Visual del Computador [en línea], [citado el 20 de Octubre del 2007]. Disponible es: < <http://www.compumedicina.com/oftalmologia/svc.pdf> >
- [10] Schlossberg, Eric B., Morrow, Sandra, Llosa, Augusto E., Mamary Edward, Dietrich, Peter y Rempel, David M. 2004. Upper Extremity Pain and Computer Use Among Engineering Graduate Students. *American journal of industrial medicine*. 46:297-303.
- [11] University of Wales, Centre for Intelligent Systems-Department of Computer Science. Creating a Case-Base using CASL and An Introductory Guide to Caspian. http://www.aber.ac.uk/~dcswwww/Research/mbsg/cbrprojects/getting_caspian.shtml
- [12] Katz N., Amick B., Carol B., Hollis C., Fossel A. y Coley C. 2000. Prevalence of upper extremity musculoskeletal disorders in college students. *Am J Med* 109:586-588.
- [13] Juan Manuel Fernández Vázquez, Javier Camacho Galindo, Ortopedistas Hospital ABC. Dolor en muñeca y mano (segunda parte). Clínica, Dolor Revista Mexicana de Algología y Terapia, VOL. 5, No. 3, MARZO-ABRIL 2007. ISSN 1665-3238. (<http://www.intramed.net/>).
- [14] Juan Manuel Fernández Vázquez y Javier Camacho Galindo, Ortopedistas Hospital ABC. Dolor en muñeca y mano (primera parte). Clínica, Dolor Revista Mexicana de Algología y Terapia, VOL. 4, No. 2, Nov 2007. ISSN 1665-3238. (<http://www.intramed.net/>).
- [15] J.M. Garcia Gorrostieta, S. Gonzalez Lopez, Sistema Ergonómico para Laboratorios de Computo, Primer Congreso de Ergonomía y Seguridad Industrial (2007), Universidad de la Sierra División de Ingenierías y Tecnologías. isbn: 978-968-9488-00-2.
- [16] Pantallas de visualización de datos (P.V.D.): fatiga postural. Maria Félix Villar Fernández (Licenciada en Ciencias Biológicas), Pedro A. Begueria Latorre. CENTRO NACIONAL DE NUEVAS TECNOLOGÍAS, Notas Técnicas de Prevención visto en Julio'08 de http://empleo.mtin.es/insht/ntp/ntp_232.htm.

Knowledge Representation and Logic

This page intentionally left blank

Probabilistic Dynamic Belief Logic for Image and Reputation

Isaac Pinyol^a Jordi Sabater-Mir^a Pilar Dellunde^{a,b}

^a *IIIA-Artificial Intelligence Research Institute*

CSIC-Spanish Scientific Research Council

Bellaterra, Barcelona, Spain

{ipinyol,jsabater,pilar}@iiia.csic.es

^b *Univ. Autònoma de Barcelona*

Bellaterra, Barcelona, Spain

pilar.dellunde@uab.cat

Abstract. Since electronic and open environments became a reality, computational trust and reputation models have attracted increasing interest in the field of multi-agent systems (MAS). Some of them are based on cognitive theories of reputation that require cognitive agents to display all their potential. One of them is Repage, a computational system based on a cognitive theory of reputation that focuses on the study of two important concepts: Image and Reputation. The possible integration of these Repage predicates into a cognitive agent architecture, like the well-known *Belief, Desire, Intention* (BDI) approach implies the definition of these two concepts as mental states, as a set of beliefs. In this paper, we specify a belief logic that captures the necessary elements to express Image and Reputation and we study their interplay by analyzing a classical definition of trust on information sources.

Keywords. Image, Reputation, Probabilistic Dynamic Belief Logic, Multi-agent systems

1. Introduction

Computational trust and reputation models have been recognized as one of the key technologies required to design and implement agent systems [12]. These models provide evaluations of the agents' performances towards specific situations (social evaluations) that agents might use to select partners. In recent years, many models have been developed [14,13], but two main approaches currently exist in the literature. On the one hand, centralized approaches consider reputation and trust to be a global and public property of the agent. They are widely used in online web sites such as eBay, Amazon, etc. On the other hand, distributed approaches consider reputation and trust to be a subjective property of each agent. In this case, this system becomes an important part of the agents architecture.

One of these models is Repage [15], a computational system based on a cognitive theory of reputation [3] that describes a model of REputation and imAGE. Although both are social evaluations, image and reputation are distinct objects. Image is a simple eval-

uative belief; it tells that a target agent is *good* or *bad* with respect to a norm, a standard or a skill. Reputation is a belief about the existence of a communicated evaluation.

Repage model entails a tight integration with the agent architecture to exploit all its potential. In particular, cognitive agent architectures will allow the agent to reason not only about the final value of trust or reputation but also about all the individual elements that contribute to that value.

This work is a first step in this direction. A tight integration of Repage model with a cognitive agent must start with the representation as mental states of the main Repage predicates: Image and Reputation. To do so in Section 2 we introduce the concept of social evaluation within the context of Repage model. In Section 3 we define the BC logic that we use as a belief logic. In Section 4 we propose a possible description of Repage image and reputation predicates in terms of our BC logic. In Section 5 we study a condition that makes coincide image and reputation mental states of the agents. In Section 6 we state some of the related work regarding trust formalizations, and finally, we conclude in Section 7 by exposing the conclusions and future work.

2. Social Evaluations in Repage: Image and Reputation

A social evaluation is a generic term used to cover the information referring to the evaluation that a social entity might have about the performance, regarding some skill, standard or norm, of another social entity. A social entity is an active member of the society, like a single agent, a group of agents or institutions.

As we mentioned in the introduction, Repage provides social evaluations as image (what agent believes) and reputation (what agents say). Previous works already show the importance of keeping this distinction [3]. Repage builds images from direct experiences and communicated images from other agents, and reputation only from communicated reputation. The influence between them is done, at the moment, at the pragmatic-strategic level of the agent, letting the agent decide which source of information to use.

In Repage, social evaluations are a simplification of the generic view given in [3]. All of them have an owner of the evaluation, a target agent (the agent being evaluated), a role (the context of the evaluation that encapsulates the behavior being evaluated) and the value of the evaluation (how *good* or *bad*).

The role is the object of the evaluation, the context. The evaluation of an agent playing the role of *buyer* can be totally different from playing the role of *car driver*. This concept of role is similar to the one used in electronic institutions [7]. The value of the evaluation is represented with a tuple of five elements, showing a probability distribution over the labels *Very Bad*, *Bad*, *Neutral*, *Good*, *Very Good* ($\{VB, B, N, G, VG\}$ from now on). So, the sum of all values is exactly 1.

Image and reputation predicates are represented as follows:

- $Img_i(t, r, [V_{VB}, V_B, V_N, V_G, V_{VG}])$
- $Rep_i(t, r, [V_{VB}, V_B, V_N, V_G, V_{VG}])$

For instance, the predicate $Img_i(t, seller, [0.5, 0.3, 0.2, 0, 0])$ indicates that agent i evaluates agent t as a seller, and with a probability of 0.5 she acts as *very bad*, with a 0.3, as *bad* and with a 0.2, as *neutral*. Since this social evaluation is an image, this represents what agent i believes. With the same evaluation but as reputation

$Rep_i(t, seller, [0.5, 0.3, 0.2, 0, 0])$ indicates that agent i believes that the evaluation circulates in the society.

In Repage, the role embraces two pieces of information. On one side, it determines the evaluation function, the mapping between the result of a transaction and the sorted set of labels $\{VB, B, N, G, VG\}$, and on the other side, it determines which actions are required to be executed by the agent that intends to evaluate another agent by the specific role. For instance, if the role seller is evaluated with the quality of the attribute obtained after the transaction *buy*, and this quality goes from 0 to 100, we could define *VB* as qualities between 0 and 20, *B* as qualities between 20 and 40, and so on.

In the next section we describe a logical framework to express in terms of beliefs these two concepts, allowing then formal reasoning.

3. Defining BC-Logic

3.1. Introduction

The logic we introduce in this section (*BC*-logic) is a probabilistic dynamic logic with a set of special modal operators B_i , S_{ij} and S_i expressing what is believed by agent i , what has been said from agent i to agent j and what has been said by agent i to however, respectively. The dynamic aspect of this logic is introduced by defining a set Π of actions. Then, for $\alpha \in \Pi$ and $\varphi \in BC$, formulas like $[\alpha]\varphi$ indicate that after the execution of α , the formula φ holds. A very important characteristic of this logic is the inclusion of the special action of communicating a formula. In this case, if φ is a formula and i, j are agents, the expression $[\varphi_{ij}]\phi$ indicates after the communication of φ from agent i to agent j , the formula ϕ holds.

Our language allows explicit reasoning about probability of formulas by means of a new operator Pr , representing the probability of holding a formula. Then, for formulas $\varphi, \phi \in BC$ the expression $Pr\varphi \leq Pr\phi$ indicates that the probability of holding φ is smaller or equal to the probability of holding ϕ . Furthermore, we allow one side of the inequality to be a constant \bar{r} where $r \in [0, 1] \cap \mathbb{Q}$. Our language is based on the Logic of Knowledge and Probability introduced by Fagin and Halpern in [8].

BC-logic allows to express formulas like $B_i(0.8 \leq Pr([\alpha]\varphi))$, meaning that agent i believes that the probability of holding φ after the execution of action α is at least 0.8. Thereby, the formula $S_i(0.8 \leq Pr([\alpha]\varphi))$ expresses the same but in terms of what agent i has said. Notice that there is not a necessary implication between both concepts. Like in human societies, people might communicate information that they do not believe.

3.2. The Syntax

To introduce *BC*-logic we start by defining a countable set P of propositional variables, a finite set Π_0 of atomic programs and a finite set \mathcal{A} of agent identifiers of cardinality n . The set of formulas $Fm(BC)$ and the set Π of programs are defined in Backus-Naur extended form as follows:

$$\begin{aligned} \phi ::= & p \mid \top \mid \neg\phi \mid \phi \wedge \varphi \mid [\alpha]\phi \mid \\ & B_i\phi \mid S_{ij}\phi \mid Pr\varphi \leq Pr\phi \mid \bar{r} \leq Pr\phi \mid Pr\varphi \leq \bar{r} \end{aligned}$$

and

$$\alpha ::= \pi \mid \alpha; \beta \mid \alpha \cup \beta \mid \alpha^*$$

where $p \in P$, $\pi \in \Pi_0$, ϕ, φ are formulas, α, β are programs, i, j agent's identifiers and $r \in [0, 1] \cap \mathcal{Q}$.

From now on we assume that a subset $\Pi_1 \subseteq \Pi_0$ of the set of atomic actions are a fixed set of speech acts. Thus, our language contains formulas of the following form $[\varphi_{ij}]\phi$ where φ_{ij} is a *BC*-formula.

We write $\varphi \vee \phi$ as an abbreviation for $\neg(\neg\varphi \wedge \neg\phi)$, $\varphi \rightarrow \phi$ as an abbreviation for $\varphi \vee \neg\phi$, $\varphi \leftrightarrow \phi$ for $(\varphi \rightarrow \phi) \wedge (\phi \rightarrow \varphi)$, \perp as an abbreviation for $\neg\top$. Finally let $S_i\phi$ be a short cut for $\bigvee_{j \in \mathcal{A}} S_{ij}\phi$ and $S\varphi$ a short cut for $S_{i_1}\varphi \wedge \dots \wedge S_{i_n}\varphi$ where $\{i_1, \dots, i_n\} = \mathcal{A}$. We write $a = b$ as an abbreviation of $a \leq b \wedge b \leq a$.

Also, we would like to point out that this language definition allows an arbitrary number of nested modal operators. When thinking about modeling agent's mind we are very interested in allowing meta-beliefs, so, beliefs on others' beliefs. In this way, agents can model other agents' minds. Following the same idea, why not allowing agents modeling other agents' minds but from the point of view of another third agent? This would require three-level nested beliefs. Even if in real applications more than three levels could seem not useful, from a theoretical point of view we want to keep this possibility open.

3.3. The Semantics

We want to give to the logic a probabilistic interpretation. For this, semantics of *BC*-logic are given by means of Kripke structures of the following form $M = \langle W, \mathcal{F}, \{R_\alpha : \alpha \in \Pi\}, \{R_{B_i} : i \in \mathcal{A}\}, e, \zeta, \mathcal{C} \rangle$ where:

- W is a non-empty set of possible worlds.
- $\{R_\alpha : \alpha \in \Pi\}$ are the accessibility relations for actions, for each $\alpha \in \Pi$, $R_\alpha \in 2^{(W \times W)}$.
- $\{R_{B_i} : i \in \mathcal{A}\}$ are the accessibility relations for the belief operators, for each agent i , $R_{B_i} \in 2^{(W \times W)}$.
- $e : P \rightarrow 2^W$ assigns to each propositional variable a set of worlds.
- $\mathcal{C} : W \rightarrow 2^{\Pi_1}$ assigns to each world a repository of *BC*-formulas.
- $\zeta : W \rightarrow PS$ assigns a probability space to each world.

where PS is the class of all probability spaces $(W, \mathcal{G}_w, \mu_w)$ such that \mathcal{G}_w is the field of all subsets of W and the function $\mu_w : 2^W \rightarrow [0, 1]$ is a finitely additive probability measure on 2^W , that is, $\mu_w(\emptyset) = 0$, $\mu_w(W) = 1$ and whenever $X, Y \in 2^W$ and $X \cap Y = \emptyset$ we have $\mu_w(X \cup Y) = \mu_w(X) + \mu_w(Y)$.

Definition A Kripke model $M = \langle W, \mathcal{F}, \{R_\alpha : \alpha \in \Pi\}, \{R_{B_i} : i \in \mathcal{A}\}, e, \zeta, \mathcal{C} \rangle$ is *regular* iff for every $\alpha, \beta \in \Pi$: (1) $R_{\alpha;\beta} = R_\alpha \circ R_\beta$, (2) $R_{\alpha \cup \beta} = R_\alpha \cup R_\beta$ and (3) $R_{\alpha^*} = R_\alpha^*$ (where $*$ is the ancestral of the relation R_α).

From now on we will only consider regular Kripke models as semantics for our logic. Let M be a model for the logic and $w \in W$ a possible world, given $p \in P$, $\varphi, \phi \in BC^1$, $\alpha \in \Pi$, $i, j \in \mathcal{A}$ and $r \in [0, 1] \cap \mathcal{Q}$, the truth-value of a *BC*-formula on model M is defined as follows:

¹We will write *BC* instead of $Fm(BC)$ to refer to the set of all well-formed formulas

- $M, w \models \top$
- $M, w \models p$ iff $w \in e(p)$
- $M, w \models \varphi \wedge \phi$ iff $M, w \models \varphi$ and $M, w \models \phi$
- $M, w \models \neg\varphi$ iff $M, w \not\models \varphi$
- $M, w \models B_i\varphi$ iff $\forall w_k : (w, w_k) \in \mathcal{R}_{B_i}$ implies $M, w_k \models \varphi$
- $M, w \models S_{ij}\varphi$ iff $\varphi_{ij} \in \mathcal{C}(w)$
- $M, w \models [\alpha]\varphi$ iff $\alpha \in \Pi$ and $\forall w_k : (w, w_k) \in \mathcal{R}_\alpha$ implies $M, w_k \models \varphi$
- $M, w \models Pr\varphi \leq \bar{r}$ iff $\mu_w(\{w_k | M, w_k \models \varphi\}) \leq r$
- $M, w \models \bar{r} \leq Pr\phi$ iff $r \leq \mu_w(\{w_k | M, w_k \models \phi\})$
- $M, w \models Pr\varphi \leq Pr\phi$ iff $\mu_w(\{w_j | M, w_j \models \varphi\}) \leq \mu_w(\{w_k | M, w_k \models \phi\})$

The semantics of formulas of the form $S_{ij}\varphi$ is introduced by means of the function \mathcal{C} , that assigns to every world the set of sentences communicated among the agents up to this moment. Then, $S_{ij}\varphi$ is true if and only if agent i has said φ to somebody. Notice that $[\varphi_{ij}]$ is a modal operator but $S_{ij}\varphi$ is not².

Notice that in the semantics we assign a probability space for each world. This is the most general case. In certain conditions and depending on the context we can force a semantic condition stating that all worlds have the same probability space. Nevertheless, in the context of reputation models we are interested in keeping the general case, since some knowledge about probability distributions may depend on the information that the reputation model offers, and this information is totally dynamic and may change.

3.4. Axiomatization

We present now some axioms for BC -logic although our purpose is not to provide a complete axiomatization. Our first axioms and rules are those of classical propositional dynamic logic plus the standard axioms $K, D, 4$ and 5 of modal logic and necessitation rules for the B_i operators.

- BK: $B_i(\varphi \rightarrow \phi) \rightarrow (B_i\varphi \rightarrow B_i\phi)$
- BD: $B_i\varphi \rightarrow \neg B_i\neg\varphi$
- B4: $B_i\varphi \rightarrow B_iB_i\varphi$
- B5: $\neg B_i\varphi \rightarrow B_i\neg B_i\varphi$

Regarding the operators S_{ij} , we include the following axioms:

- S1: $[\varphi_{ij}]S_{ij}\varphi$
- S2: $S_{ij}\varphi \rightarrow B_jS_{ij}\varphi$
- S3: $S_{ij} \rightarrow [\alpha]S_{ij}$, for every $\alpha \in \Pi$

For probabilistic formulas, we include the following axioms:

- P1: $Pr\top = 1$
- P2: $Pr\perp = 0$
- P3: $0 \leq Pr\varphi$
- P4: $Pr((\varphi \wedge \phi) = a) \wedge Pr(\varphi \wedge \neg\phi) = b \rightarrow Pr\varphi = a + b$
- P5: $B_i(Pr\varphi = a) \wedge B_i(Pr(\varphi \rightarrow \phi) = b) \rightarrow B_i(max(a + b - 1, 0)) \leq Pr\phi$
- P6: $Pr\varphi = Pr\phi$, if $\varphi \leftrightarrow \phi$ is a theorem.

²For the sake of clarity we have written formulas like $S_{ij}\varphi$ and $Pr\varphi \leq \bar{r}$ instead of $S_i[\varphi]$ and $Pr[\varphi] \leq r$. We have not make explicit that formulas in this context are reified, avoiding the use of many-sorted languages

$P1$ and $P2$ state the probabilities of \top and \perp . $P3$ claims for the non-negativity of probability formulas. $P4$ is the additivity axiom. $P5$ is the equivalent Lukasiewicz implication for multimodal logic for beliefs and probabilistic formulas. In fact this axiom can be deduced from the previous ones, but we make it explicit because it is very useful for the reasoning process we will achieve in the future work. $P6$ is distributivity. As inference rule we include also: from φ it can be derived $Pr\varphi = 1$.

4. Grounding Image and Reputation

At this point our interest relies on giving to Repage predicates a description in terms of the BC logic we introduced in Section 3. However, we need to introduce first, the nomenclature we will use in the Repage domain.

4.1. Some Notation

Having the finite set \mathcal{A} of agent identifiers and the finite set \mathcal{R} of role identifiers, the actions are determined by the possible roles and possible agents. As a matter of simplification, we assume that each role r has associated only one generic action, $\Phi(r)$, that at a certain moment of time T may be executed to some agent j , $\Phi(r)[j]_T$. Also, Repage encapsulates the way agents evaluate outcomes from transactions. In this sense, the evaluation of an agent as a seller can be determined, for instance, by the quality of the product obtained and the delivery time. The proposition $\delta_T(r).q$ will refer to the value of the attribute q of the role r obtained in transaction T .

To illustrate this, we could define $\Phi(seller)$ as the action of *buy* and $\Phi(seller)[John]$ as the action of buying to John. We will write it as *Buy(John)*. Then, if the evaluation of the role *seller* is done thorough the attributes quality and delivery time, we will write the proposition $\delta_T(seller).quality$ and $\delta_T(seller).deliveryTime$ to refer to the respective values.

4.2. Image and Reputation Predicates

Let i, j be agent identifiers and r a role Repage image and reputation predicates are represented as $Img_i(j, r, [V_{VB}, V_B, V_N, V_G, V_{VG}])$ and $Rep_i(j, r, [V_{VB}, V_B, V_N, V_G, V_{VG}])$. Agent i is the agent that has generated the predicate, and therefore, that uses Repage. Agent j is the target of the evaluation. Vector $[V_{VB}, V_B, V_N, V_G, V_{VG}]$ represents the five probabilistic values that cover the full space of possible outcomes, which are classified as *Very Bad*, *Bad*, *Neutral*, *Good* and *Very Good*. Following the definition of Image, we have that a Repage image and reputation predicates can be expressed with the following set of beliefs written in BC logic:

| Image | Reputation |
|---|--|
| $B_i(V_{VB} = Pr([\Phi(r)[j]]\Psi_{VB}))$ | $B_i(S(V_{VB} = Pr([\Phi(r)[j]]\Psi_{VB})))$ |
| $B_i(V_B = Pr([\Phi(r)[j]]\Psi_B))$ | $B_i(S(V_B = Pr([\Phi(r)[j]]\Psi_B)))$ |
| ... | ... |

The expression Ψ_X is the propositional formula that depends on the specific role r and that evaluates the possible outcome obtained after the execution of action $\Phi(r)[t]$ (the conditions for which an outcome is classified as *Very Bad*, *Bad* etc...). For instance, is Repage has generated the predicate $Img_i(S1, seller, [.5, .2, .1, .1, .1])$, and the role

seller is evaluated with the attribute quality (from 0 to 100) obtained after the transaction *buy*, the set of beliefs describing the mental state of the agent regarding seller *S1* could be:

- $B_i(0.5 = Pr([Buy(S1)]0 \leq \delta(seller).quality < 20))$
- $B_i(0.2 = Pr([Buy(S1)]20 \leq \delta(seller).quality < 40))$
- ...

If the agent is cognitive, these beliefs participate in the deliberation process. Notice that this mental state does not say anything about which potential seller is *better*. This will be determined by the set of desires. Notice that dealing with more or less condition levels (other than five: VB,...,VG) would not represent a big change.

5. Image, Reputation and Their Interplay

One of the key points of Repege and the cognitive theory of reputation that underlies it [3] is the relationship between image and reputation. The theory states that both are social evaluations but distinct objects. With the representation we give for image and reputation in *BC* logic this difference depends on the relationship between the belief operator *B* and the operator *S*. As a matter of fact, the inclusion of axioms relating both concepts would generate a typology of agents. We discuss some of them in the following subsections.

5.1. Honest and Consistent Agents

Let $i \in \mathcal{A}$, we say that the agent *i* is honest if the formula $S_i\varphi \rightarrow B_i\varphi$ is included in her theory (when she says something, she believes it), and we abbreviate it as $h_i\varphi$. In a similar way, a consistent agent *i* will hold in her theory the formula $S_i\varphi \rightarrow \neg S_i\neg\varphi$ (when she says something, she never says the opposite) abbreviated as $c_i\varphi$. It is easy to prove that $h_i\varphi \rightarrow c_i\varphi$.

The definition of honesty allows agents to model what other agents think (meta-beliefs) in terms of what they say. In this sense, if the formula $B_i(h_j\varphi)$ holds (agent *i* believes that agent *j* is honest), applying definitions and axiom *K* for *B* operator we obtain $B_iS_j\varphi \rightarrow B_iB_j\varphi$.

5.2. Trusting Agents

The concept of trust has different connotations and many definitions. From a cognitive point of view, trust can be seen as a mental state [2] of a particular agent that believes that another agent has certain property [6]³. Considering trust on agents as information sources, Demolombe in [6] introduces six definitions of trust: Sincerity, Credibility, Vigilance, Validity and Completeness. The following definition of trust coincide with the Validity definition, that at the same time, is a conjunction between Sincerity and Credibility:

³This vision of trust is somehow a simplification of the general view given in [2] where trust is considered a mental attitude composed of beliefs and goals and strongly related to the concept of delegation.

Let i, j be agents, we define $Trust_{i \rightarrow j} \varphi$ as $B_i(S_j \varphi \rightarrow \varphi)$. The formula states that agent i trusts agent j when agent i believes that whatever j says is true. Applying axiom K , the formula becomes $B_i S_j \varphi \rightarrow B_i \varphi$. This means that if agent i believes that j has said something, agent i will believe the same thing. This notion of trust is very strong. In fact, the generalization of this formula to an agent that *trusts* everybody, make collapse the definition of image and reputation we gave for Repage predicates.

Proposition 5.1 *Let i be an agent, if $B_i(S\varphi \rightarrow \varphi)$ holds, then the beliefs describing reputation predicates from Repage collapse with the beliefs describing image predicates.*

Proof The proof is quite direct. Applying axiom K to $B_i(S\varphi \rightarrow \varphi)$ we obtain $B_i S\varphi \rightarrow B_i \varphi$. The antecedent of the implication coincides with the definition of reputation predicates we gave. Since φ is an arbitrary formula, applying modus ponens to each one of the formulas used to describe reputation predicates, for instance, $B_i(S(V_{VB} = Pr([\Phi(r)[j]]\Psi_{VB})))$ we obtain $B_i(V_{VB} = Pr([\Phi(r)[j]]\Psi_{VB}))$. The set of all these new beliefs coincide with the definition of image that we gave.

This result states that we have a condition that makes logically equivalent the mental state of an agent holding an image and a reputation. Notice that the condition $B_i(\varphi \rightarrow S\varphi)$ collapses image mental state with reputation mental state.

6. Related Work

Some current state-of-the-art logics inspired us for defining the BC logic. However, none of them seems to be expressive enough for the needs we have described in this paper. The probabilistic and dynamic notions have been mostly treated in epistemic logic ([10], [8]), and in a simpler way in belief logic [1]. Furthermore, some formalizations of trust using belief logic have been done [11], where trust is related to information acquisition in multi-agent systems, but in a crisp way. Similar to this, in [5], modal logic is used to formalize trust in information sources, also with crisp predicates. Here, actions and communicated formulas are also used.

Regarding fuzzy reasoning on trust issues, in [9] it is defined a trust management system in a many-valued logic framework where beliefs are graded. Also, in [4] it is proposed a logic that integrates reasoning about graded trust (on information sources) and belief fusion in multi-agent systems. Our logic does not use graded beliefs. Instead, we use the notion of beliefs on probability sentences, because when deal with image and reputation it seems a more accurate option.

7. Conclusion and future work

In this work we have introduced a probabilistic dynamic belief logic to capture the mental states of agents holding image and reputation predicates as defined in Repage model. The logic seems to be expressive enough to describe them and to provide a logical framework in which to define a typology of agents.

In a short period of time, our plans include the study and tentative proof of completeness and soundness of the axiomatization we have given for BC logic. Also, we plan to

study more conditions that make image and reputation influence each other at the level of beliefs. In particular, we are very interested in the redefinition of the *trust* predicate by including a grade: $Trust_{i \rightarrow j}^g \varphi$ as $B_i(Pr(S_j \varphi \rightarrow \varphi) = g)$. Also, we plan to study this relationship across the reminding definitions of trust: *Vigilance*, *Cooperativeness*, *Completeness* and the combination of them.

We plan to use this logic as a fundamental part of a BDI agent where desires, intentions and plans are build taking as a base this logic. The importance of the *BC* logic relies on that from this moment on, we have a logical framework that allows us to express all what we need referring to social evaluations.

8. Acknowledgments

This work was supported by the European Community under the FP6 programme (eRep project CIT5-028575 and OpenKnowledge project FP6-027253), by the projects Autonomic Electronic Institutions (TIN2006-15662-C02-01) and Agreement Technologies (CONSOLIDER CSD2007-0022, INGENIO 2010) and by the Generalitat de Catalunya, under the grant 2005-SGR-00093. Also, we would like to give special thanks to Robert Demolombe for his useful comments.

References

- [1] A. Casali, Ll. Godo, and C. Sierra. Graded models for bdi agents. In J. Leite and P. Torroni, editors, *CLIMA V, Lisboa, Portugal*, pages 18–33, 2004. ISBN: 972-9119-37-6.
- [2] C. Castelfranchi and R. Falcone. Social trust. In *Proceedings of the First Workshop on Deception, Fraud and Trust in Agent Societies, Minneapolis, USA*, pages 35–49, 1998.
- [3] R. Conte and M. Paolucci. *Reputation in artificial societies: Social beliefs for social order*. Kluwer Academic Publishers, 2002.
- [4] R. Demolombe and C.J. Liau. A logic of graded trust and belief fusion. In *Proc. of the 4th Workshop on Deception, Fraud and Trust in Agent Societies*, pages 13–25, 2001.
- [5] R. Demolombe and E. Lorini. A logical account of trust in information sources. In *Eleventh International Workshop on Trust In Agent Societies*, 2008.
- [6] Robert Demolombe. To trust information sources: a proposal for a modal logical framework. *Trust and deception in virtual societies*, pages 111–124, 2001.
- [7] M. Esteva, J. A. Rodríguez-Aguilar, C. Sierra, P. Garcia, and J. L. Arcos. On the formal specifications of electronic institutions. In *Agent Mediated Electronic Commerce, The European AgentLink Perspective.*, pages 126–147. Springer-Verlag, 2001.
- [8] R. Fagin and J.Y. Halpern. Reasoning about knowledge and probability. *J. ACM*, 41(2):340–367, 1994.
- [9] T. Flaminio, G. Michele Pinna, and E. B. P. Tiezzi. A complete fuzzy logical system to deal with trust management systems. *Fuzzy Sets Syst.*, 159(10):1191–1207, 2008.
- [10] B. P. Kooi. Probabilistic dynamic epistemic logic. *J. of Logic, Lang. and Inf.*, 12(4):381–408, 2003.
- [11] C. J. Liau. Belief, information acquisition, and trust in multi-agent systems: a modal logic formulation. *Artif. Intell.*, 149(1):31–60, 2003.
- [12] M. Luck, P. McBurney, O. Shehory, and S. Willmott. *Agent Technology: Computing as Interaction (A Roadmap for Agent Based Computing)*. AgentLink, 2005.
- [13] S.D. Ramchurn, D. Hunyh, and N.R. Jennings. Trust in multi-agent systems. *The Knowledge Engineering Review*, 1(19):1–25, 2004.
- [14] J. Sabater and C. Sierra. Review on computational trust and reputation models. *Artif. Intel. Rev.*, 24(1):33–60, 2005.
- [15] J. Sabater-Mir, M. Paolucci, and R. Conte. Repage: Reputation and image among limited autonomous partners. *J. of Artificial Societies and Social Simulation*, 9(2), 2006.

Aggregation Operators and Ruled Surfaces

J. Recasens

Sec. Matemàtiques i Informàtica. ETS Arquitectura del Vallès. UPC

e-mail: j.recasens@upc.edu

Abstract. Aggregation operators in two variables that are ruled quadric surfaces are studied.

The interest lays in the fact that the most popular aggregation operators are indeed ruled surfaces, either planes or quadric surfaces.

Keywords. Aggregation operator, Idempotent, Symmetric, Ruled quadric surface

Introduction

The most popular aggregation operators (in two variables), namely arithmetic, geometric, quadratic and harmonic means, OWA operators, the Minimum t-norm and Maximum t-conorm, are ruled surfaces. Geometric, quadratic and harmonic means are quadric surfaces while the graphics of the other ones consist of piecewise plane surfaces.

This paper studies other ruled quadric surfaces that correspond to aggregation operators. In this way new families of aggregation operators, some of them combinations of the previous ones, are obtained.

Let us recall the definition of aggregation operator (in two variables).

Definition 0.1. [1] *An aggregation operator (in two variables) is a map $h : [0, 1]^2 \rightarrow [0, 1]$ satisfying for all $x, y, x_1, x_2, y_1, y_2 \in [0, 1]$*

1. $h(0, 0) = 0$ and $h(1, 1) = 1$
2. $h(x_1, y_1) \leq h(x_2, y_2)$ if $x_1 \leq x_2$ and $y_1 \leq y_2$ (monotonicity).

h is idempotent if and only if $h(x, x) = x$. h is symmetric if and only if $h(x, y) = h(y, x)$.

1. Ruled Quadric Surfaces

Definition 1.1. *A quadric surface is a surface defined in implicit form by a second degree polynomial*

$$ax^2 + by^2 + cz^2 + dxy + exz + fyz + gx + hy + iz + j = 0. \quad (1)$$

In order to find the ruled quadric surfaces which are aggregation operators, we will consider separately the cases $c \neq 0$ and $c = 0$.

1.1. Case $c = 0$

If $c = 0$, then isolating z from Eq. (1) we obtain

$$z = -\frac{ax^2 + by^2 + dxy + gx + hy + j}{ex + fy + i}. \quad (2)$$

Replacing e by $-e$, f by $-f$ and i by $-i$, Eq. (2) is

$$z = \frac{ax^2 + by^2 + dxy + gx + hy + j}{ex + fy + i}.$$

If we want the last map to be symmetric, we must have $b = a$, $g = h$ and $e = f$, obtaining

$$z = \frac{ax^2 + ay^2 + dxy + gx + gy + j}{ex + ey + i}.$$

$z(0, 0)$ must be 0. From this we have $j = 0$.

If z is idempotent, writing explicitly $z(x, x) = x$ we obtain

$$z(x, x) = \frac{2ax^2 + dx^2 + 2gx}{2ex + i} = x$$

or

$$(2a - 2e + d)x^2 = (i - 2g)x.$$

This equation is satisfied for all $x \in [0, 1]$ if and only if

$$d = 2e - 2a.$$

and

$$i = 2g.$$

The formula of the quadric surface becomes then

$$z = \frac{ax^2 + ay^2 + (2e - 2a)xy + gx + gy}{ex + ey + 2g}. \quad (3)$$

Now we can consider two cases: $e \neq 0$ and $e = 0$.

1.1.1. Case $c = 0$ and $e \neq 0$

In this case we can divide the numerator and the denominator of Eq. (3) by e . Renaming $\frac{a}{e}$ by a , and $\frac{g}{e}$ by g , we get

$$z = \frac{a(x-y)^2 + 2xy + gx + gy}{x + y + 2g}.$$

The denominator must be different from 0 for all $x, y \in (0, 1)$. This means

$$g \geq 0 \text{ or } g \leq -1.$$

For $x = 0$ and $y = 1$, we obtain

$$z(0, 1) = \frac{a + g}{1 + 2g}$$

This value must be between 0 and 1. Imposing that it must be greater or equal than 0, we obtain the following conditions for a and g .

$$g \geq -\frac{1}{2} \text{ and } a \geq -g$$

or

$$g \leq -\frac{1}{2} \text{ and } a \leq -g$$

Imposing that it must be smaller or equal than 1, we obtain the following conditions for a and g .

$$g \geq -\frac{1}{2} \text{ and } a \leq g + 1$$

or

$$g \leq -\frac{1}{2} \text{ and } a \geq g + 1$$

The partial derivatives $\frac{\partial z}{\partial x}(1, 0)$ and $\frac{\partial z}{\partial x}(0, 1)$ must be greater or equal than 0.

$$\frac{\partial z}{\partial x}(1, 0) = \frac{(2a + g)(1 + 2g) - a - g}{(1 + 2g)^2} \geq 0$$

is satisfied if and only if

$$g \geq -\frac{1}{4} \text{ and } a \geq \frac{-2g^2}{1 + 4g}$$

or

$$g \leq -\frac{1}{4} \text{ and } a \leq \frac{-2g^2}{1+4g}.$$

$$\frac{\partial z}{\partial x}(0,1) = \frac{(-2a+g+2)(1+2g)-a-g}{(1+2g)^2} \geq 0$$

is satisfied if and only if

$$g \geq -\frac{3}{4} \text{ and } a \leq \frac{2+2g^2+4g}{3+4g}$$

or

$$g \leq -\frac{3}{4} \text{ and } a \geq \frac{2+2g^2+4g}{3+4g}.$$

Sumarizing, the conditions on g and a are

$$g \geq 0 \text{ and } \frac{-2g^2}{1+4g} \leq a \leq \frac{2+2g^2+4g}{3+4g}$$

or

$$g \leq -1 \text{ and } \frac{2+2g^2+4g}{3+4g} \leq a \leq \frac{-2g^2}{1+4g}.$$

1.1.2. Case $c = 0$ and $e = 0$

In this case, putting $\frac{a}{2g} = b$,

$$z = b(x-y)^2 + \frac{x+y}{2}.$$

$z(1,0)$ is then

$$b + \frac{1}{2}.$$

Imposing again that this value must be between 0 and 1, we get that

$$-\frac{1}{2} \leq b \leq \frac{1}{2}.$$

Imposing that the partial derivative $\frac{\partial z}{\partial x}(1,0)$ must be greater or equal than 0, we get

$$b \geq -\frac{1}{4}.$$

Imposing that the partial derivative $\frac{\partial z}{\partial x}(0,1)$ must be greater or equal than 0, we get

$$b \leq \frac{1}{4}.$$

Summarizing,

$$-\frac{1}{4} \leq b \leq \frac{1}{4}.$$

or

$$g \leq 0 \text{ and } \frac{g}{2} \leq a \leq -\frac{g}{2}.$$

1.2. Case $c \neq 0$

If $c \neq 0$, we can divide Eq. (1) by c . Renaming $\frac{a}{c}$ by $-a$, $\frac{b}{c}$ by $-b$, etc, we obtain

$$z = \frac{1}{2} \left(ex + fy + i \pm \sqrt{(ex + fy + i)^2 - 4ax^2 - 4by^2 - 4gx - 4dxy - 4hy - 4j} \right).$$

If we impose symmetry we get

$$z = \frac{1}{2} \left(ex + ey + i \pm \sqrt{(e(x + y) + i)^2 - 4ax^2 - 4ay^2 - 4gx - 4dxy - 4gy - 4j} \right). \quad (4)$$

We can distinguish the cases where the square root is added or subtracted.

1.2.1. Adding the square root

In this case, Eq. (4) becomes

$$z = \frac{1}{2} \left(ex + ey + i + \sqrt{(e(x + y) + i)^2 - 4ax^2 - 4ay^2 - 4gx - 4dxy - 4gy - 4j} \right).$$

Imposing $z(0, 0) = 0$, we get

$$i + \sqrt{i^2 - 4j} = 0$$

and therefore $i \leq 0$ and $j = 0$.

From $z(1, 1) = 1$, we get

$$2 = 2e + i + \sqrt{(2e + i)^2 - 8a - 8g - 4d}$$

and from this, $1 - 2e - i + 2a + 2g + d = 0$.

From $z(\frac{1}{2}, \frac{1}{2}) = \frac{1}{2}$, we get

$$1 = e + i + \sqrt{(e + i)^2 - 2a - 4g - 2d}$$

and from this, $1 - 2e - 2i + 2a + 4g + d = 0$.

So $i = 2g$ (and $g \leq 0$) and $d = -1 + 2e - 2a$.

Now imposing $z(k, k) = k$ we get

$$k = \frac{1}{2} \left(2ek + i + \sqrt{(2ke + i)^2 - 8ak^2 - 8gk - 4dk^2} \right).$$

which is equivalent to

$$\begin{aligned} k &= \frac{1}{2} \left(2ek + i + \sqrt{(2ke + i)^2 - 8ek^2 - 4ik + 4k^2} \right) \\ &= \frac{1}{2} \left(2ek + i + \sqrt{(2ke + i - 2k)^2} \right). \end{aligned}$$

Then

$$2ke + i \leq 2k \text{ for all } k \in [0, 1].$$

This is satisfied for all $k \in [0, 1]$ if and only if

$$2e + i \leq 2. \quad (5)$$

Putting $b = \frac{e}{2}$, the equation of the quadric surface is then

$$z = b(x + y) + g + \sqrt{(b(x + y) + g)^2 - a(x - y)^2 - g(x + y) + (1 - 4b)xy}.$$

and Eq. 5 becomes

$$2b + g \leq 1.$$

$$z(1, 0) = b + g + \sqrt{(b + g)^2 - a - g}$$

which implies

$$g + a \leq (b + g)^2.$$

$0 \leq z(1, 0) \leq 1$ gives

$$b + g \leq 1, g + 2b - a \leq 1$$

and if $b + g \leq 0$, then

$$a + g \leq 0.$$

Now imposing that $\frac{\partial z}{\partial x}(1, 0) \geq 0$ we obtain

$$b + \frac{1}{2} \frac{2b^2 + 2bg - 2a - g}{\sqrt{(b + g)^2 - a - g}} \geq 0$$

and $\frac{\partial z}{\partial x}(0, 1) \geq 0$ gives

$$b + \frac{1}{2} \frac{2b^2 + 2bg + 2a - g + 1 - 4b}{\sqrt{(b + g)^2 - a - g}} \geq 0.$$

1.2.2. Subtracting the square root

$z(0, 0) = 0$ implies

$$j = 0 \text{ and } i \geq 0.$$

$z(1, 1) = 1$ implies

$$1 - 2e - i + 2a + 2g + d = 0 \quad (6)$$

and $z(\frac{1}{2}, \frac{1}{2}) = \frac{1}{2}$

$$1 - 2e - 2i + 2a + 4g + d = 0. \quad (7)$$

From Eqs. (6) and (7), we get

$$i - 2g = 0 \text{ (and } g \geq 0)$$

and

$$d = -1 + 2e - 2a.$$

Now imposing $z(k, k) = k$ we get

$$k = \frac{1}{2} \left(2ek + i - \sqrt{(2ke + i)^2 - 8ak^2 - 8gk - 4dk^2} \right).$$

which is equivalent to

$$\begin{aligned} k &= \frac{1}{2} \left(2ek + i - \sqrt{(2ke + i)^2 - 8ek^2 - 4jk + 4k^2} \right) \\ &= \frac{1}{2} \left(2ek + i - \sqrt{(2ke + i - 2k)^2} \right). \end{aligned}$$

Then

$$2ke + i \geq 2k \text{ for all } k \in [0, 1].$$

This inequality is satisfied for all $k \in [0, 1]$ if and only if

$$2e + i \geq 2.$$

Putting $b = \frac{e}{2}$, the equation of the quadric surface is then

$$z = b(x + y) + g - \sqrt{(b(x + y) + g)^2 - a(x - y)^2 - g(x + y) + (1 - 4b)xy}.$$

and Eq. 1.2.2 becomes

$$2b + g \geq 1.$$

$$z(1, 0) = b + g - \sqrt{(b + g)^2 - a - g}.$$

From this,

$$a + g \leq (b + g)^2.$$

$0 \leq z(1, 0) \leq 1$ gives

$$g + a \geq 0$$

and if $b + g \geq 1$, then

$$g + 2b - a \geq 1.$$

Now imposing that $\frac{\partial z}{\partial x}(1, 0) \geq 0$ we obtain

$$b - \frac{1}{2} \frac{2b^2 + 2bg - 2a - g}{\sqrt{(b + g)^2 - a - g}} \geq 0.$$

From $\frac{\partial z}{\partial x}(0, 1) \geq 0$ we obtain

$$b - \frac{1}{2} \frac{2b^2 + 2bg + 2a - g + 1 - 4b}{\sqrt{(b + g)^2 - a - g}} \geq 0.$$

2. Concluding remarks

The quadric ruled surfaces that can be considered as idempotent and symmetric aggregation operators have been studied.

Table 1 summarizes the results obtained in this work.

In forthcoming works, other aggregation operators such as t-norms, t-conorms or uninorms whose graphics are quadric ruled surfaces will be studied (see [2]),

- If in the first equation $a = g = 0$, we recover the harmonic mean.
- If in the second equation $b = 0$, we recover the arithmetic mean.
- If in the third equation $b = g = a = 0$ we recover the geometric mean.
- If in the third equation $a = -\frac{1}{2}$ and $b = g = 0$, we recover the quadratic mean.
- If in the third equation $a = g = 0$ and $b = \frac{1}{2}$, we recover the Maximum aggregation operator.
- If in the fourth equation $a = g = 0$ and $b = \frac{1}{2}$, we recover the Minimum aggregation operator.
- If $\frac{1}{2} \leq p \leq 1$ and in the third equation $a = g = 0$ and $b = \frac{p}{2}$, we recover the OWA operator with weights p and $1 - p$.
- If $0 \leq p \leq \frac{1}{2}$ and in the fourth equation $a = g = 0$ and $b = \frac{p}{2}$, we recover the OWA operator with weights p and $1 - p$.

Table 1. Ruled quadric surfaces that are idempotent and symmetric aggregation operators

| | |
|---|--|
| 1 | $z = \frac{a(x-y)^2 + 2xy + gx + gy}{x+y+2g}$ $g \geq 0 \text{ and } \frac{-2g^2}{1+4g} \leq a \leq \frac{2+2g^2+4g}{3+4g}$ <p>or</p> $g \leq -1 \text{ and } \frac{2+2g^2+4g}{3+4g} \leq a \leq \frac{-2g^2}{1+4g}$ |
| 2 | $z = b(x-y)^2 + \frac{x+y}{2}$ $-\frac{1}{4} \leq b \leq \frac{1}{4}$ |
| 3 | $z = b(x+y) + g + \sqrt{(b(x+y)+g)^2 - a(x-y)^2 - g(x+y) + (1-4b)xy}$ $g \leq \min(0, 1-2b, 1-b)$ $-1+g+2b \leq a \leq (b+g)^2 - g$ $b + \frac{1}{2} \frac{2b^2+2bg-2a-g}{\sqrt{(b+g)^2-a-g}} \geq 0$ $b + \frac{1}{2} \frac{2b^2+2bg+2a-g+1-4b}{\sqrt{(b+g)^2-a-g}} \geq 0$ <p>If $b+g \leq 0$, then $a+g \leq 0$.</p> |
| 4 | $z = b(x+y) + g - \sqrt{(b(x+y)+g)^2 - a(x-y)^2 - g(x+y) + (1-4b)xy}$ $g \geq \max(0, 1-2b, -b)$ $a \leq \min((b+g)^2 - g, 1+g+2b)$ $b - \frac{1}{2} \frac{2b^2+2bg-2a-g}{\sqrt{(b+g)^2-a-g}} \geq 0$ $b - \frac{1}{2} \frac{2b^2+2bg+2a-g+1-4b}{\sqrt{(b+g)^2-a-g}} \geq 0$ <p>If $b+g \geq 1$, then $g+2b-a \geq 1$.</p> |

Acknowledgements

Research partially supported by project number TIN2006-14311.

References

- [1] G. Beliakov, A. Pradera, T. Calvo. (2008) Aggregation Functions: A Guide for Practitioners. Studies in Fuzziness and Soft Computing.
- [2] C. Alsina, A. Sklar. (1987). A characterization of continuous associative operations whose graphs are ruled surfaces. *Aequationes Mathematicae* **33** 114-119.

New Challenges: Group Decision Systems by Means of Entropy Defined through Qualitative Reasoning Structures¹

Llorenç Roselló,^{a,2} Francesc Prats,^a Mónica Sánchez^a and Núria Agell^b

^a Polytechnical University of Catalonia, Barcelona, Spain

^b ESADE, Ramon Llull University, Barcelona, Spain

Abstract. This paper introduces the concept of entropy on orders of magnitude qualitative spaces and, consequently, the opportunity to measure the gain or loss of information when working with qualitative descriptions. The results obtained are significant in terms of situations which arise naturally in many real applications when dealing with different levels of precision. A method that enables ambiguous values to be included in analysis and permits summarizing information can be defined. The entropy allows measuring the consensus of expert opinion by use of a distance.

Keywords. Qualitative Reasoning, Magnitude Order Spaces, Entropy, Information

Introduction

Qualitative Reasoning (QR) is a sub-area of Artificial Intelligence that seeks to understand and explain human beings' ability to reason qualitatively [2], [5]. The main objective is to develop systems that allow decisions to be taken where numerical data are either lacking or insufficient for this purpose. As indicated in [11], this could be due to both a lack of information as well as to an information overload.

A main goal of Qualitative Reasoning is to tackle problems in such a way that the principle of relevance is conserved. In other words, each variable has to be valued with the requisite level of precision [3]. It is not unusual for a situation to arise in which one needs to work simultaneously with various levels of precision, depending on the available information, in order to ensure interpretability of the results obtained. The mathematical structures of Orders of Magnitude Qualitative Spaces (S) were introduced to this end.

This paper makes two contributions. First of all, a way of measuring the amount of information of a given system when it is described by means of orders of magnitude is presented. To this end, the study focuses on defining entropy for orders of magnitude qualitative spaces. Secondly, such entropy allows one to consider conditional entropy,

¹This work has been partly funded by MEC (Spanish Ministry of Education and Science) AURA project (TIN2005-08873-C02).

²Corresponding Author: llorenç.rosello@upc.edu, Dept. de Matemàtica Aplicada 2, UPC

which in turn permits the induced distance to be used in optimising decision-making problems.

Recently, the concepts presented in this paper have been applied to group decision-making problems in business settings, where experts' opinions are considered from a collaborative standpoint and are used to select and rank a set of alternatives based on ordinal information [9].

However, the entropy concept given in this paper could also be applied in weighing up the views of a group of experts or of a large group of staff. By using web-based technologies, the system presented could be extended to improve forecasting in a business context - for example, in new fad-driven markets where the collective wisdom of the work force is used to forecast market trends.

Retail marketing furnishes an example of the entropy concept facilitating qualitative reasoning with orders of magnitude. The information provided by a set of expert managers on several features describing a retailing firm is considered. Features are evaluated by each manager on an ordinal scale with different levels of precision and from two points of view: (1) the feature's importance for the firm's success; (2) the current performance of the feature. In this context, the entropy would help measure either the information quality of the experts' opinions or the relevance of the considered features, thus helping managers make the right decisions to improve the firm's performance.

The concept of entropy has its origins in the nineteenth century, particularly in thermodynamics and statistics. This theory has been developed from two points of view: the macroscopic (Carnot, Clausius, Gibbs, Planck and Caratheodory), and the microscopic (developed by Maxwell and Boltzmann) [8]. In particular, the statistical concept of Shannon's entropy, related to the microscopic approach, is a measure of the amount of information [10],[1]. In this paper the concept of entropy considered was inspired by Hillman's work.

The paper is organised in the following way. Section 1 presents the qualitative models of absolute orders of magnitude, and Section 2 provides these models with a semi-lattice structure. Section 3 defines axiomatically entropy in these qualitative spaces following Hillman's theory, and the distance induced by this entropy. In Section 4, the natural generalisation of the introduced concepts in the space of multi-dimensional qualitative labels is performed. After the theoretic development of the methodology presented, in Section 5, an example of effective calculation of the entropy in a particular case is carried out; it allows interpreting the results in terms of quantity of information of qualitative descriptions. Lastly, conclusions and open problems are presented.

1. The Absolute Orders of Magnitude Model §

The absolute orders of magnitude model ([6]; [7]) works with a structure defined via a finite set of qualitative ordered labels. A general algebraic structure, called Qualitative Algebra or Q-algebra, was defined on this framework [15],[10], providing a mathematical structure which unifies sign algebra and interval algebra through a continuum of qualitative structures built from the rougher to the finest partition of the real line.

The one-dimensional absolute orders of magnitude model [11] works with a finite number of qualitative labels corresponding to an ordinal scale of measurement. The number of labels chosen to describe a real problem is not fixed, but depends on the characteristics of each represented variable.

Let's consider a finite set of *basic* labels $\mathbb{S}_* = \{B_1, \dots, B_n\}$, which is totally ordered as a chain: $B_1 < \dots < B_n$. Depending on the number of considered basic labels, each basic label can correspond to a linguistic term, as for instance “extremely bad” < “very bad” < “bad” < “acceptable” < “good” < “very good” < “extremely good”. It is not unusual the case when the basic labels are defined by a discretization given by a set $\{a_1, \dots, a_{n+1}\}$ of real numbers as landmarks, then $B_i = [a_i, a_{i+1}]$, $i = 1, \dots, n$.

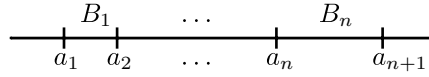


Figure 1: Basic labels via discretization

Nevertheless, in this paper, the most general case has been considered, in which the knowledge of the values of the landmarks is not required to introduce the basic labels.

The *non-basic* label $[B_i, B_j]$, with $i < j$, is defined as the set:

$$[B_i, B_j] = \{B_i, B_{i+1}, \dots, B_j\}.$$

In the case in which B_i is the real interval $[a_i, a_{i+1}]$, then the label $[B_i, B_j]$ corresponds to the interval $[a_i, a_{j+1}]$.

The complete universe of description for the Orders of Magnitude Space $\text{OM}(n)$, with granularity n , where n is the number of basic labels, is the set \mathbb{S} :

$$\mathbb{S} = \mathbb{S}_* \cup \{[B_i, B_j] \mid B_i, B_j \in \mathbb{S}_*, i < j\}.$$

Using the convention $[B_i, B_i] = B_i$, the set \mathbb{S} is:

$$\mathbb{S} = \{[B_i, B_j] \mid B_i, B_j \in \mathbb{S}_*, i \leq j\}.$$

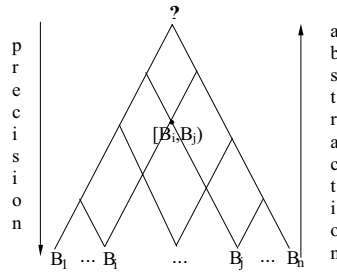


Figure 2. The space \mathbb{S}

Following the former example of linguistic labels, the label “moderately good” can be represented by [“acceptable”, “good”], i.e. $[B_4, B_5]$, and the label “don’t know” is represented by [“extremely bad”, “extremely good”], i.e. $[B_1, B_7]$.

2. The Semi-Lattice (\mathbb{S}, \vee)

Let Λ be the set that represents a magnitude or a feature that is qualitatively described by means of the labels of \mathbb{S} . Since Λ can represent both a continuous magnitude such as

position and temperature, etc., and a discrete feature such as salary and colour, etc., Λ could be considered as the range of a function

$$a : I \subset \mathbb{R} \rightarrow Y,$$

$\Lambda = \{a(t) = a_t \mid t \in I\}$, where Y is a convenient set. For instance, if a is a room temperature during a period of time $I = [t_0, t_1]$, Λ is the range of temperatures during this period of time. Another example can be considered when $I = \{1, \dots, n\}$ and $\Lambda = \{a(1), \dots, a(n)\}$, where n is the number of people whose eye colour we aim to describe. The process of qualitativization is given by a function

$$Q : \Lambda \rightarrow \mathbb{S},$$

where $a_t \mapsto Q(a_t) = E_t = \text{minimum label (with respect to the inclusion } \subset) \text{ which describes } a_t$, i.e. the most precise qualitative label describing a_t . All the elements of the set $Q^{-1}(E_t)$ are “representatives” of the label E_t or “are qualitatively described” by E_t . They can be considered qualitatively equal.

The function Q induces a partition in Λ by means of the equivalence relation:

$$a \sim_Q b \iff Q(a) = Q(b).$$

This partition will be denoted by Λ / \sim_Q , and its equivalence classes are the sets $Q^{-1}(Q(a_j)) = Q^{-1}(E_j)$, $\forall j \in J \subset I$. Each of these classes contains all the elements of Λ which are described by the same qualitative label.

Let E_1, E_2 be the labels describing qualitatively the elements of $Q^{-1}(E_1)$ and $Q^{-1}(E_2)$ respectively; then the *join* operation between E_1 and E_2 , $E_1 \vee E_2$, is defined as the minimum label that describes the elements $Q^{-1}(E_1)$ and the elements $Q^{-1}(E_2)$, i.e. the label that describes the set $Q^{-1}(E_1) \cup Q^{-1}(E_2)$; it is the convex union of the labels E_1 and E_2 . The convex union of two labels $[B_i, B_j], [B_h, B_k] \in \mathbb{S}$ has to be understood as the label $[B_{\min\{i, h\}}, B_{\max\{j, k\}}]$.

Proposition 1 (\mathbb{S}, \vee) is a semi-lattice.

Proof It is trivial to check that \vee in \mathbb{S} is idempotent, commutative and associative. \square

The couple (\mathbb{S}, \vee) is called *joinset* in [4]

The *join* operation induces a partial order relation in \mathbb{S} as follows: let be $E_1, E_2 \in \mathbb{S}$, then $E_1 \leq E_2$ iff $E_1 \vee E_2 = E_2$. If $E_1 = [B_i, B_j], E_2 = [B_h, B_k]$, then $E_1 \leq E_2 \iff E_1 \vee E_2 = E_2 \iff \min\{i, h\} = h, \max\{j, k\} = k \iff h \leq i, j \leq k \iff E_1 \subset E_2$, that is, E_1 is more precise than E_2 .

3. An Entropy on \mathbb{S}

3.1. Axiomatic Definition of Entropy

In this section a measure of the amount of information will be defined. Unlike most of magnitudes as the mass, distance, energy, etc, in the case of information there is no device that can measure it. To measure it there is a mathematical concept, the *entropy*. The approach followed in this paper to define an *entropy*, a *conditional entropy* and a *distance* induced by the entropy is inspired by the work of Hillman (see [4]).

An entropy is defined on a semi-lattice (Ω, \vee) by means of a function $H : \Omega \rightarrow \mathbb{R}$ that verifies the following axioms [4]:

1. *Positivity*: $H(x) \geq 0 \forall x \in \Omega$.
2. *Monotonicity*: If $x \leq y$, then $H(x) \leq H(y)$.
3. *Contractivity*: If $x \leq y$, then $H(y \vee z) - H(x \vee z) \leq H(y) - H(x) \forall z \in \Omega$.

Proposition 2 *The function H defined on \mathbb{S} by $H([B_i, B_j]) = \text{card}(\{B_i, \dots, B_j\}) = j - i + 1$ is an entropy on (\mathbb{S}, \vee) .*

Proof Positivity and monotonicity are trivially satisfied.

The third axiom of contractivity: $[E_1 \leq E_2 \implies H(E_2 \vee E_3) - H(E_1 \vee E_3) \leq H(E_2) - H(E_1) \forall E_3]$ is straightforwardly checked.

In effect, let $E_1 = [B_i, B_j]$, $E_2 = [B_h, B_k]$, $E_3 = [B_r, B_s]$, with $E_1 \leq E_2$. Then $h \leq i \leq j \leq k$. The formula that has to be proved is:

$(\max\{k, s\} - \min\{h, r\} + 1) - (\max\{j, s\} - \min\{i, r\} + 1) \leq (k - h + 1) - (j - i + 1)$, which is equivalent to

$(\max\{k, s\} - \max\{j, s\}) + (\min\{i, r\} - \min\{h, r\}) \leq (k - j) + (i - h)$.

It suffices to check that $\max\{k, s\} - \max\{j, s\} \leq k - j$ and $\min\{i, r\} - \min\{h, r\} \leq i - h$. This is immediate taking into account that $j \leq k$ and $h \leq i$ and considering the different positions of s and r in relation with the intervals $[j, k]$ and $[h, i]$, respectively. \square

The entropy of a label $[B_i, B_j]$ is the number of basic labels contained in it.

3.2. Conditional Entropy

Given two labels E_1, E_2 , the *conditional entropy* $H(E_1/E_2)$ is:

$$H(E_1/E_2) = H(E_1 \vee E_2) - H(E_2).$$

In this way, knowing the number of basic labels which describe the elements of $Q^{-1}(E_2)$, $H(E_1/E_2)$ is the number of basic labels missing to describe the elements of $Q^{-1}(E_2) \cup Q^{-1}(E_1)$. This conditional entropy can be interpreted as the change produced to the entropy when passing from the qualitative description E_2 to the qualitative description $E_1 \vee E_2$.

3.3. The Metric Space (\mathbb{S}, \vee, D)

The *distance induced by the entropy* between two labels E_1 and E_2 is:

$$D(E_1, E_2) = H(E_1/E_2) + H(E_2/E_1) = 2H(E_1 \vee E_2) - H(E_1) - H(E_2).$$

This expression satisfies the axioms of a distance, and will be useful for decision-making and optimization processes based on the amount of information.

4. The Case of Multi-Dimensional Labels

Let $\mathbb{S}_1, \dots, \mathbb{S}_n$ be n orders of magnitude spaces with corresponding granularities m_1, \dots, m_n and sets of basic labels $\mathbb{S}_{1*}, \dots, \mathbb{S}_{n*}$.

The set of n -dimensional labels is defined as:

$$\mathbb{S}_1 \times \dots \times \mathbb{S}_n = \{ \mathbf{E} = (E^1, \dots, E^n) \mid E^i \in \mathbb{S}_i, i = 1, \dots, n \},$$

where the n -dimensional label $\mathbf{E} = (E^1, \dots, E^n)$ is a vector of n qualitative labels, each of them eventually belonging to different orders of magnitude spaces.

Considering the join operation

$$\mathbf{E} \vee \mathbf{E}' = (E^1, \dots, E^n) \vee (E'^1, \dots, E'^n) = (E^1 \vee E'^1, \dots, E^n \vee E'^n),$$

the semi-lattice in this case is $(\mathbb{S}_1 \times \dots \times \mathbb{S}_n, \vee)$, because associativity, commutativity and idempotency in each \mathbb{S}_i induce the same properties in the cartesian product $\mathbb{S}_1 \times \dots \times \mathbb{S}_n$.

The partial order \leq in this semi-lattice is given by:

$$\mathbf{E} = (E^1, \dots, E^n) \leq \mathbf{E}' = (E'^1, \dots, E'^n) \iff \mathbf{E} \vee \mathbf{E}' = \mathbf{E}',$$

which is equivalent to $E^i \vee E'^i = E'^i$, that is to say, $E^i \leq E'^i$ for all $i = 1, \dots, n$.

Proposition 3 The function H defined on $\mathbb{S}_1 \times \dots \times \mathbb{S}_n$ by

$$H(\mathbf{E}) = H(E^1, \dots, E^n) = H(E^1) + \dots + H(E^n)$$

is an entropy on $(\mathbb{S}_1 \times \dots \times \mathbb{S}_n, \vee)$.

Proof Positivity is trivially satisfied. With respect to monotonicity, $\mathbf{E} = (E^1, \dots, E^n) \leq \mathbf{E}' = (E'^1, \dots, E'^n) \implies E^i \leq E'^i \forall i \implies H(E^i) \leq H(E'^i) \forall i \implies H(\mathbf{E}) = \sum H(E^i) \leq \sum H(E'^i) = H(\mathbf{E}')$.

The third axiom of contractivity:

$$\mathbf{E} \leq \mathbf{E}' \implies H(\mathbf{E}' \vee \mathbf{E}'') - H(\mathbf{E} \vee \mathbf{E}'') \leq H(\mathbf{E}') - H(\mathbf{E}) \forall \mathbf{E}'' = (E''^1, \dots, E''^n)$$

is also satisfied; in effect, for any $\mathbf{E}'' = (E''^1, \dots, E''^n)$:

$$\mathbf{E} \leq \mathbf{E}' \implies E^i \leq E'^i \forall i \implies H(E'^i \vee E''^i) - H(E^i \vee E''^i) \leq H(E'^i) - H(E^i) \forall i \implies \sum H(E'^i \vee E''^i) - \sum H(E^i \vee E''^i) \leq \sum H(E'^i) - \sum H(E^i) \implies H(\mathbf{E}' \vee \mathbf{E}'') - H(\mathbf{E} \vee \mathbf{E}'') \leq H(\mathbf{E}') - H(\mathbf{E}). \quad \square$$

The conditional entropy is given in this case, by

$$H(\mathbf{E}/\mathbf{E}') = H(\mathbf{E} \vee \mathbf{E}') - H(\mathbf{E}'),$$

which is equal to $\sum H(E^i/E'^i)$.

Finally, the distance induced by the entropy is defined by

$$D(\mathbf{E}, \mathbf{E}') = H(\mathbf{E}/\mathbf{E}') + H(\mathbf{E}'/\mathbf{E}),$$

which is equal to $\sum D(E^i, E'^i)$. This is the Manhattan distance in the set $\mathbb{S}_1 \times \dots \times \mathbb{S}_n$.

Next, an example of effective calculation of the entropy and the distance presented is carried out in a particular case, which allows interpreting the results in terms of quantity of information of qualitative descriptions.

5. An Example

Let us consider four ordinal measures of performance perceived by customers in a supermarket: 1) Store renovation, 2) Store location, 3) Staff training and 4) The quality of private brand products

And let us consider the evaluations of these four features provided by two experts with different levels of precision in an ordinal scale with seven basic labels: $\mathbb{S}_* = \{B_1, \dots, B_7\}$, with: B_1 =“extremely bad”, B_2 =“very bad”, B_3 =“bad”, B_4 =“acceptable”, B_5 =“good”, B_6 =“very good”, and B_7 =“extremely good”.

The features are then represented via a 2-dimensional qualitative label in the set $\mathbb{S} \times \mathbb{S}$, being the first component the evaluation of the first expert and the second the evaluation of the second expert. Suppose that the features 1, 2, 3, 4 are the following:

$$\mathbf{E}_1 = ([B_1, B_7], [B_5, B_7])$$

$$\mathbf{E}_2 = ([B_2, B_4], [B_3, B_4])$$

$$\mathbf{E}_3 = (B_2, [B_3, B_5])$$

$$\mathbf{E}_4 = ([B_4, B_6], B_6)$$

as represented in Figure 3.

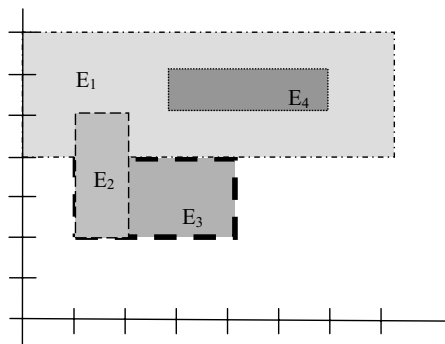


Figure 3. Representation of the 2-labels

Their entropies are: $H(\mathbf{E}_1) = 10$, $H(\mathbf{E}_2) = 5$, $H(\mathbf{E}_3) = 4$, $H(\mathbf{E}_4) = 4$.

The first feature, because of the highest entropy, is the least precise one and so the one which has less information from the two experts. Therefore, if one feature has to be discarded it will be this one. In an application to a real case, when computing with a large number of features and a large number of experts of a retailer firm, the entropy together with a landmark can indicate which features provide less information and therefore can be discarded.

The distances between the features are:

$$D(\mathbf{E}_1, \mathbf{E}_2) = 2(7 + 5) - 10 - 5 = 9, \quad D(\mathbf{E}_1, \mathbf{E}_3) = 2(7 + 5) - 10 - 4 = 10.$$

$$D(\mathbf{E}_1, \mathbf{E}_4) = 2(7 + 3) - 10 - 4 = 6, \quad D(\mathbf{E}_2, \mathbf{E}_3) = 2(3 + 3) - 5 - 4 = 3.$$

$$D(\mathbf{E}_2, \mathbf{E}_4) = 2(5 + 4) - 5 - 4 = 9, \quad D(\mathbf{E}_3, \mathbf{E}_4) = 2(5 + 4) - 4 - 4 = 10.$$

The distance measures the similarity between evaluations and supposing \mathbf{E}_1 discarded, the labels with least distance are \mathbf{E}_2 and \mathbf{E}_3 , which correspond to more similar opinions and, therefore, that eventually more will be taken into account. Again in an ap-

plication to a real case, these distances measure the similarity between experts opinions and therefore can provide a measure of consensus.

6. Conclusion and Future Research

This paper introduces the concept of entropy by means of absolute orders of magnitude qualitative spaces. This entropy measures the amount of information of a system when using orders of magnitude descriptions to represent it. Future work will consider the entropy so defined to measure the degree of agreement reached in group decision problems. This will permit dynamic assessment of a group's tendency to reach consensus in selection processes. Standard Orders of Magnitude algorithms do not apply the entropy concept at present; development of such extensions is also an important issue for future research.

A web-based software tool capable of gathering and summarising opinions and to work simultaneously with different levels of precision is being built using the concepts presented in this paper. The software developed will provide corporations with a useful tool for selecting and ranking alternatives in a collaborative way. In other words, the tool will permit consideration not only of the information given by a group of experts but also of the information provided by a larger group of people in the company. This approach should prove particularly valuable in assessing new ideas and/or trends in innovation processes. The outcome should be enhanced staff collaboration in companies, leading to more accurate forecasts and faster innovation.

References

- [1] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications, 1991.
- [2] K. Forbus. *Qualitative Reasoning*. CRC Hand-book of Computer Science and Engineering. CRC Press, 1996.
- [3] Kenneth Forbus. Qualitative process theory. *Artificial Intelligence*, 24:85–158, 1984.
- [4] Chris Hillman. A formal theory of information: 1. statics. Technical report, Department of Mathematics, University of Washington, Seattle.
- [5] B. Kuipers. Making sense of common sense knowledge. *Ubiquity*, 45(4), 1996.
- [6] F. Guerrin Luise Travé-Massuyès, P. Dague. *Le Raisonnement Qualitatif pour les pour les Sciences de l'Ingenieur*. Hermès, 1997.
- [7] Núria Piera. Current trends in qualitative reasoning and applications. Technical report, International Center for Numerical Methods in Engineering, 1995.
- [8] V.A. Rokhlin. Lectures on the entropy of measure preserving transformations. *Russian Math. Surveys*, 22:1 – 52, 1967.
- [9] Mónica Sánchez, Yu-Chiang Hu, Francesc Prats, Xari Rovira, Josep M. Sayeras, and John Dawson. Ranking features by means of qualitative optimisation process. Technical report, Departament de Matemàtica Aplicada II, Universitat Politècnica de Catalunya, 2007.
- [10] Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379 – 423, 1948.
- [11] Louise Travé-Massuyès and Philippe Dague, editors. *Modèles et raisonnements qualitatifs*. Hermes Science (Paris), 2003.

Machine Learning

This page intentionally left blank

Voltage Sag Source Location From Extracted Rules Using Subgroup Discovery

Victor BARRERA^a, Beatriz LÓPEZ^a, Joaquim MELÉNDEZ^a, Jorge SÁNCHEZ^b

^a*eXiT, UdG, Spain¹*

^b*Endesa, Barcelona, Spain²*

Abstract. This work presents a set of rules to determine the voltage sag source location in electric power systems. The rule set is extracted using subgroup discovery (SD). The SD objective is to discover characteristics of subgroups with respect to a specific property of interest. Our interest is to obtain the origin of sag events, upstream or downstream from the measurement point. Voltage sag features registered in electric substations are used as input data to SD algorithm. The SD algorithm used is CN2-SD to learn descriptive rules. Results show the rules extracted can be easily interpreted by a domain expert, allowing the formulation of heuristic classification rules with high accuracy.

Index Terms. CN2-SD, Fault location, machine learning, power quality monitoring, subgroup discovery, voltage sag (dip) source location.

1. Introduction

According to the IEEE 1159, voltage sag is considered a short duration disturbance defined by a reduction of the RMS (Root Mean Square) voltage between 0.1 and 0.9 per unit (p.u) of the nominal voltage and duration between half cycle (10 milliseconds) and one minute, **Fig. 1** shows two voltage sag events registered at electric power network.

Due to the impact on sensitive industrial loads and costs led by the damages and maintenance costs, voltage sags have focused the attention of power quality studies leaded by the utilities. Unfortunately, those disturbances propagate through the power system affecting loads connected in the whole network. Hence, the responsibility for the generation of disturbances must be assessed, being the automatic location of their origin one of the most interesting aspects involved. Given a register, two basic steps have to be accomplished: the first one is to isolate the origin of the sag, distinguishing from upstream (**Fig. 1a**) or downstream (**Fig. 1b**) localization according to the power

¹ eXiT Group in the Institute of Informatics and Applications (IIIA) of the University of Girona, Spain, Girona, Campus Montilivi, 17071, e-mail: [vbarrera, blopez, quimmel][@eia.udg.edu](mailto:[vbarrera, blopez, quimmel]@eia.udg.edu). The Spanish government under the project DPI2006-09370, grant BES-2007-14942 and ENDESA DISTRIBUCIÓN support this research.

² PQ Department of Endesa Distribución, Barcelona, Spain, e-mail: jslosada@fecsa.es.

flow direction (see Fig. 2); next an appropriate algorithm has to be used to accurately locate the origin in the network.

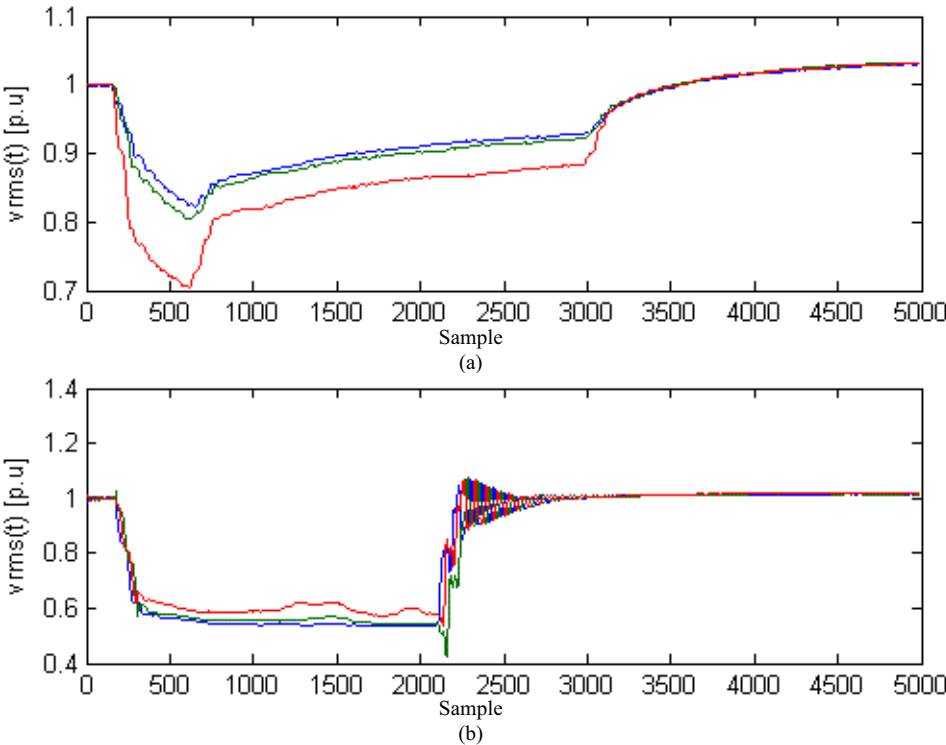


Fig. 1. Example of Voltage sag events. (a) Upstream origin, (b) Downstream origin. RMS phase voltage

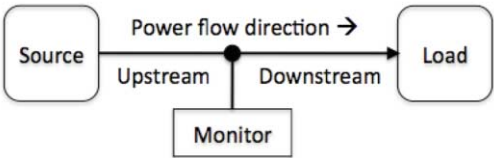


Fig. 2. Voltage sag source location

This work proposes the application of subgroup discovery to voltage sag source location. Subgroup discovery consists on learning individual rules which define interesting patterns in data. It is a learning method that is often used for preliminary exploratory data analysis. We apply with the objective of obtaining rules which describe relationships between voltage sag features and its origin (upstream or downstream). These rules can help the electric facilities to determine the responsibility for the generation of sags.

This paper is organized as follows. First, we give some basics on knowledge discovery. Next, in section 3 we explain how this methodology has been applied to our electricity domain. We continue on section 4 by reporting the results obtained as so far. We end with some conclusions and discussion.

2. Subgroup Discovery

Subgroup discovery has been defined as a task at the intersection of predictive and descriptive induction [1]. On the first group, there are supervised methods as the classification rule learning ones, that try to discover rules for solving classification problems. On the latter, the goal is to discover individual rules describing interesting regularities in data. Most of the methods are unsupervised, as the association rule learning. Subgroup discovery is an exception, since it discovers rules from labeled data.

The objective of subgroup discovery is to discover characteristics of subgroups ('most interesting individuals) with respect to a specific property of interest (represented in the rule consequent) [1]. Conversely to classification, different rules induced by a subgroup method can cover the same set of examples, while the overall set of rules left some of the examples uncovered. The aim of subgroup descriptions is to cover a significant proportion of positive examples. Algorithms as EXPLORA [2], MIDOS [3] and SubGroupMiner [4] are examples of this relatively new way of addressing a learning problem.

In subgroup discovery, rules R_i have the form $Cond_i \rightarrow Class_j$ where the property of interest for subgroup discovery is class value $Class_j$ that appears in the rule consequent, and the rule antecedent $Cond_i$ is a conjunction of features (attribute-value pairs) selected from the features describing the training instances [1].

A determining factor in the quality of any subgroup discovery algorithm is in the quality assessment of both the rules and the results of applying the procedure. Below, the CN2-SD and quality measures used in this work are described.

2.1 Subgroup Discovery Quality Measures

The following five quality indices are used to evaluate the quality of subgroup discovery rules:

- *Coverage for a rule*: Measures the percentage of examples covered on average by one of the induced set of rules [5]. It is defined as follows:

$$Cov(R_i) = p(Cond_i) = \frac{n(Cond_i)}{n_s} \quad (1)$$

Where $n(Cond_i)$ number of examples which verifies the condition $Cond_i$ (*independently* of the class), and n_s is the number of examples. The average coverage for n_r -rules is calculated as follows:

$$COV = \frac{1}{n_r} \sum_{i=1}^{n_r} Cov(R_i) \quad (2)$$

- *Support for a rule*: It is defined as the frequency of correctly classified examples covered as follows [5]:

$$Sup(R_i) = p(Class_j, Cond_i) = \frac{n(Class_j, Cond_i)}{n_s} \quad (3)$$

Where $n(Class_j, Cond_i)$ is the number of examples which satisfy the condition $Cond_i$ and also belong to the class $Class_j$. The support for n_r -rules is computed by (4):

$$SUP = \frac{1}{n_s} \sum_{j=1}^{n_c} n \left(Class_j, \bigvee_{Cond_i \rightarrow Class_j} Cond_i \right) \quad (4)$$

Where n_c is the number of class values considered, and n_s as above. The examples which satisfy several conditions are considered only once.

- *Complexity or size for a set of rules (COM)*: Complexity can be measured as the mean number of rules obtained for each class, or the mean of features per rule. The last definition was used in this paper [5].
- *Significance for a rule*: Indicates the significance of a finding, if measured by the likelihood ratio of a rule [5]. It is computed as follows:

$$Sig(R_i) = Sig(Cond_i \rightarrow Class_j) = 2 \sum_{j=1}^{n_c} \left[n(Class_j, Cond_i) \cdot \log \frac{n(Class_j, Cond_i)}{n(Class_j) \cdot p(Cond_i)} \right] \quad (5)$$

The significance for a set of rules is computed as follows:

$$SIG = \frac{1}{n_r} \sum_{i=1}^{n_r} Sig(R_i) \quad (6)$$

- *Unusualness for a rule*: Is defined as the weighted relative accuracy of a rule as follows [5]:

$$WRAcc(Cond_i \rightarrow Class_j) = \frac{n(Cond_i)}{n_s} \left(\frac{n(Class_j, Cond_i)}{n(Cond_i)} - \frac{n(Class_j)}{n_s} \right) \quad (7)$$

It can be described as the balance between the coverage of the rule ($p(Cond_i)$) and its accuracy gain ($p(Class_j, Cond_i) - p(Class_j)$). The unusualness on a set of rules is the following:

$$WRACC = \frac{1}{n_r} \sum_{i=1}^{n_r} WRAcc(R_i) \quad (8)$$

Both significance and *unusualness* measure the novelty of a subgroup. These are the most important measurement in subgroup discovery.

2.2 CN2-SD Algorithm

CN2-SD is one of the state-of-the-art method on subgroup discovery [1]. CN2-SD algorithm is implemented in KEEL³.

CN2-SD modifies the CN2 algorithm mainly by avoiding deleting the covered positive examples from the current training set. Instead, CN2-SD stores with each example a count indicating how often (with how many rules) the example has been covered so far. Initial weights (w) of all positive examples, e_j , are equal to 1 (noted as $w(e_j, 0) = 1$), which means that the example has not been covered by any rule. The examples already covered by one or more constructed rules decrease their counts according to two different schemes:

- *Multiplicative weights*: The weights decrease multiplicatively. For a given parameter $0 < \gamma < 1$, weights of covered positive examples decrease as follows: $w(e_j, i) = \gamma^i$, where $w(e_j, i)$ is the weight of example e_j being covered by i rules. If $\gamma = 1$ would result in finding the same rule over and over again, whereas with $\gamma = 0$ the algorithm would perform the same as the standard CN2 covering algorithm [1].
- *Additive weights*: The weights of covered positive examples decrease according to the formula $w(e_j, i) = (i + 1)^{-1}$. The weight of each example is inversely proportional to their coverage by previously induced rules [1].

3. Subgroup Discovery Applied to Voltage Sag Source Location

In [6]-[8] are described several methods to determine the origin of voltage sags with respect to a measurement point. The interest of this work is to propose one rule or set of rules that allow determining the origin of sags (upstream or downstream, Fig. 2). From a set of sags, these rules can be obtained applying Subgroup Discovery. In this section, we will describe the features used to represent each voltage sag. These set of features are the input to the subgroup discovery algorithm (CN2-SD). Then, in the next section we explain the rules obtained by mean of Subgroup Discovery. These rules will help determining the responsibility for the generation of voltage sags in electric power networks.

ENDESA⁴ provided the set of sag events registered in three substations (25kV) classified as upstream or downstream (100 and 73 respectively). We selected 8 features related with origin of voltage sag, some features were taken of the literature and other were proposed by us. They were computed for each sag using MatLab routines. The eight features are defined as follows:

- *RCV*: Minimum Remaining Complex Voltage contains information about the magnitude of sags [9]. RCV is a voltage signal that represents the three phase voltages.
- L_v : It is defined as the integral of the RMS voltage drop during the event [10]. L_v was computed from RCV instead of the three gathered voltage signals (9):

³ Data mining platform, <http://sci2s.ugr.es/keel/>

⁴ Spanish acronym of Energy distribution company of Barcelona, Spain

$$L_v = \sum_{sag} (1 - RCV_{p,u}(t)) \quad (9)$$

- *GVo*: This is a feature we proposed. It corresponds as the integral of the RMS zero sequence voltage ($V_0(t)_{rms}$) during the event normalized by the pre-fault voltage ($V_{pre-fault}$)(10).

$$GV_0 = \sum_{sag} \frac{V_0(t)_{rms}}{V_{pre-fault}} \quad (10)$$

- I_{ratio} : Ratio between the fundamental frequency component of the currents during and before of voltage sag event [11].
- T : Voltage sag event duration measured in samples.
- N_{seg} : $N_{seg}/2$ corresponds to number of sag transitory states.
- Rev : It is a resistance used for *Resistance Sign* voltage sag source location algorithm, see [12] for details about this algorithm.
- Z_{ratio} : Ratio between fault impedance and steady state impedance. This feature is used for *Distance Relay* voltage sag source location algorithm [13].

Finally, the input data are the 173-sags with 8-features each one, plus other representing the class (upstream/downstream). To use the CN2-SD algorithm, the input data were exported from Matlab® to the KEEL format. KEEL requires categorical features; hence the input data was discretized into five categories (very low, low, medium, high and very high).

4. Experimentation and results

The results obtained after applying CN2-SD to the described data a considerable amount of rules have been obtained. These rules contain useful information about the relationships between the features described above and the origin of the sags. A total of 67-rules were extracted by CN2-SD algorithm. Among them, 5 rules were selected thanks to the quality measures. After, two rules were proposed as a generalization of the previous one. In this section, we explain the steps and considerations to reduce and analyzed this amount of rules. In the discussion section the proposed rules will be explained and tested.

We defined two main scenarios, one per each weighting scheme (*multiplicative* and *additive*). In multiplicative scenario, three different γ parameter values have been tested: 0.5, 0.7, 0.9. In the additive scenario, it was not needed to introduce the γ parameter, because this scheme does not depend of it. The dataset was divided in five new dataset randomly selected [14]. The size of each one is the half size of the original dataset, so each new random dataset has 86 samples distributed 43% and 57% downstream and upstream sags as original dataset, respectively. We decided to divide the original dataset in order to do several experiment and analyze the repetitiveness of each rule. Finally, 20-experiments (15-multiplicative, 5-additive) were performed and 67 rules were generated. The next selection criteria were used to select the best experiments and reduce the amount of rules to analyze:

- The number of rules (N_R) is three or four.
- The complexity (COM) is between two or three ($2 \leq COM \leq 3$).
- The significance (SIG) and weight relative accuracy ($WRAcc$) is high.

As a result, 5-experiments were selected and the amount of rules was reduced to 17-rules (25.4%). In **Table 1** are shown the selected experiments. The first row,

according to the selection criteria, is the best experiment for $\gamma=0,5$. It contains 4 rules ($N_R=4$) and the average quality measures are $COM=2.5$, $COV=0.411$, $SUP=0.966$, $SIG=30.297$, $WRAcc=0,155$; μ , σ , σ/μ of the 4 rules appear at the bottom of each measure.

Analyzing the quality measures of **Table 1** we can observe that: (1) the *average coverage* of each experiment is $\sim 0,4$; which means that the rule set of each experiment cover $\sim 40\%$ of the input data. (2) The *support* is $\sim 0,9$; so the major part of input data are correctly classified. (3) The ratio σ/μ takes an average value equal to 14,7%, demonstrating the low variability of the quality measures in each experiment. Next, the rules contained in each experiment were analyzed. As a result, 5 rules were selected (Table 2). Rule 1 and Rule 2 were the rules most repeated in each rule set.

Table 1. Selected Experiments

| γ | N_R | COM | COV | SUP | SIG | WRAcc |
|--|----------------------|----------------------|---------------------|----------------------|---------------------|----------------------|
| Multiplicative Scheme | | | | | | |
| 0,5 | 4 | 2,5 | 0,411 | 0,966 | 30,297 | 0,155 |
| | $\mu=3.4$ | $\mu=2.867$ | $\mu=0.434$ | $\mu=0.936$ | $\mu=24,491$ | $\mu=0.131$ |
| | $\sigma=0.548$ | $\sigma=0.701$ | $\sigma=0.091$ | $\sigma=0,043$ | $\sigma=4,061$ | $\sigma=0.017$ |
| | $\sigma/\mu=16.1\%$ | $\sigma/\mu=24.4\%$ | $\sigma/\mu=21\%$ | $\sigma/\mu=4,6\%$ | $\sigma/\mu=16.6\%$ | $\sigma/\mu=13.04\%$ |
| 0,7 | 3 | 2,667 | 0,414 | 0,966 | 27,995 | 0,157 |
| | $\mu=3,2$ | $\mu=3,067$ | $\mu=0,448$ | $\mu=0,931$ | $\mu=23,226$ | $\mu=0,142$ |
| | $\sigma=0,447$ | $\sigma=0,796$ | $\sigma=0,073$ | $\sigma=0,043$ | $\sigma=3,402$ | $\sigma=0,013$ |
| | $\sigma/\mu=13,98\%$ | $\sigma/\mu=25,95\%$ | $\sigma/\mu=16,4\%$ | $\sigma/\mu=4,641\%$ | $\sigma/\mu=14,6\%$ | $\sigma/\mu=9,4\%$ |
| 0,9 | 3 | 2,667 | 0,479 | 0,897 | 28,734 | 0,161 |
| | $\mu=3,6$ | $\mu=3,267$ | $\mu=0,467$ | $\mu=0,938$ | $\mu=23,509$ | $\mu=0,152$ |
| | $\sigma=0,894$ | $\sigma=0,548$ | $\sigma=0,061$ | $\sigma=0,030$ | $\sigma=4,312$ | $\sigma=0,013$ |
| | $\sigma/\mu=24,8\%$ | $\sigma/\mu=16,8\%$ | $\sigma/\mu=13,1\%$ | $\sigma/\mu=3,2\%$ | $\sigma/\mu=18,3\%$ | $\sigma/\mu=8,9\%$ |
| 0,9 | 4 | 3 | 0,414 | 0,966 | 27,687 | 0,162 |
| | $\mu=3,6$ | $\mu=3,267$ | $\mu=0,467$ | $\mu=0,938$ | $\mu=23,509$ | $\mu=0,152$ |
| | $\sigma=0,894$ | $\sigma=0,548$ | $\sigma=0,061$ | $\sigma=0,030$ | $\sigma=4,312$ | $\sigma=0,013$ |
| | $\sigma/\mu=24,8\%$ | $\sigma/\mu=16,8\%$ | $\sigma/\mu=13,1\%$ | $\sigma/\mu=3,2\%$ | $\sigma/\mu=18,3\%$ | $\sigma/\mu=8,9\%$ |
| Additive Scheme | | | | | | |
| Additive scheme does not depend of γ | 3 | 2,667 | 0,414 | 0,966 | 27,995 | 0,150 |
| | $\mu=3,2$ | $\mu=3,067$ | $\mu=0,449$ | $\mu=0,934$ | $\mu=23,228$ | $\mu=0,132$ |
| | $\sigma=0,447$ | $\sigma=0,795$ | $\sigma=0,072$ | $\sigma=0,043$ | $\sigma=3,402$ | $\sigma=0,015$ |
| | $\sigma/\mu=13,9\%$ | $\sigma/\mu=25,9\%$ | $\sigma/\mu=16,1\%$ | $\sigma/\mu=4,6\%$ | $\sigma/\mu=14,6\%$ | $\sigma/\mu=11,3\%$ |

Table 2. Selected Rules

| Rule |
|---|
| Rule 1: IF $RCV \neq \text{High}$ AND $Rey \leq \text{Medium}$ AND $Z_{ratio} = \text{Very low}$ AND $T \neq \text{High}$ THEN downstream END |
| Rule 2: IF $I_{ratio} > \text{Very low}$ THEN downstream END |
| Rule 3: IF $RCV < \text{Very High}$ AND $I_{ratio} = \text{Very low}$ THEN upstream END |
| Rule 4: IF $GV_0 = \text{Very low}$ AND $I_{ratio} = \text{Very low}$ AND $T \neq \text{Low}$ THEN upstream END |
| Rule 5: IF $Rey \leq \text{Low}$ AND $RCV \neq \text{High}$ THEN downstream END |

5. Discussion

From Rule 2, Rule 3 and Rule 4 it is possible to observe that the upstream sags have low unbalance grade (GV_0) and low ratio current (I_{ratio}). Rule 1 shows that downstream sags are associated with medium or low Rey and very low I_{ratio} . This is very relevant information about the sag source location. However, these rules have to be generalized in order to perform a classification process. That is, the subgroup discovery rules have reveled interesting information about the sag source location, but these rules have to be generalized in order to perform the classification process. For instance the highlighted features in **Table 2** can be deleted. In doing so, a human expert can take advantage of the exploratory results provided by the subgroup discovery and propose the generalized rules shown in **Table 3**.

Table 3. Generalized Rules

| Rule |
|--|
| Rule 1: IF $Rey \leq Th_{Rey}$ AND $Z_{ratio} = Th_{Zratio}$ THEN downstream END |
| Rule 2: IF $I_{ratio} > Th_{Iratio}$ THEN downstream END |
| Rule 3: IF $R_{CV} < Th_{RCV}$ AND $I_{ratio} = Th_{Iratio}$ THEN upstream END |
| Rule 4: IF $GV_0 = Th_{GV_0}$ AND $I_{ratio} = Th_{Iratio}$ THEN upstream END |
| Rule 5: IF $Rey \leq Th_{Rey}$ THEN downstream END |

These rules depend on new discretization values, namely Th_{Rey} , Th_{Zratio} , Th_{Iratio} , Th_{RCV} and Th_{GV_0} , which can be obtained from behavior of each feature. For instance, in the plot of **Fig. 3** shows the relationship between the Z_{ratio} and Rey , in which it is possible to observe that the downstream sags are inside of area $Rey < 0$ and $Z_{ratio} < 0,95$. Therefore, Th_{Rey} can be set to 0, and Th_{Zratio} equal to 0,95. Analogously, the remaining thresholds can be extracted using the appropriate plots (**Fig. 4**, **Fig. 5**), obtaining $Th_{Iratio} = 3$, $Th_{RCV} = 0,875$ and $Th_{GV_0} = 100$.

Table 4. Confusion Matrix of the Five Rules

| Rule | TP | FN | FP | TN | SEN | SPE | 1-SPE |
|------|----|----|----|----|------|-------|-------|
| 1 | 94 | 6 | 8 | 65 | 0,94 | 0,890 | 0,110 |
| 2 | 98 | 2 | 36 | 37 | 0,98 | 0,507 | 0,493 |
| 3 | 98 | 2 | 0 | 73 | 0,98 | 1,000 | 0,000 |
| 4 | 98 | 2 | 0 | 73 | 0,98 | 1,000 | 0,000 |
| 5 | 48 | 52 | 0 | 73 | 0,48 | 1,000 | 0,000 |

Once the new thresholds have been set up, the rules of **Table 3** can be used for classification purposes. In **Table 4** are shown the confusion matrix for each rule. Rule 3 and Rule 4 have excellent results: they discriminate the sag source location from the current and voltage magnitudes. Therefore, sag event due to three-phase small motor

starting and downstream connected will be classified as upstream by these rules, because the sag will have low unbalanced grade and low fault current. To solve this problem, the current flow direction has to be taken into account. Both R_{ey} and Z_{ratio} contain information about current flow direction. In this work Z_{ratio} has been added to these rules because it discriminates the sag origin better than R_{ey} (Fig. 3).

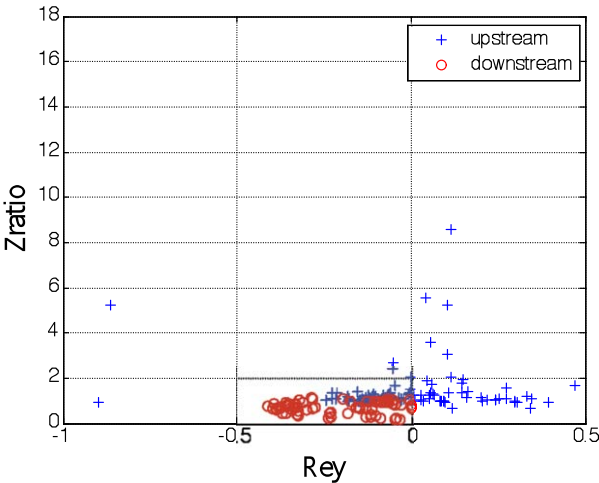


Fig. 3. R_{ey} Vs Z_{ratio} . The downstream sags are inside of area $R_{ey} < 0$ and $Z_{ratio} < 1$.

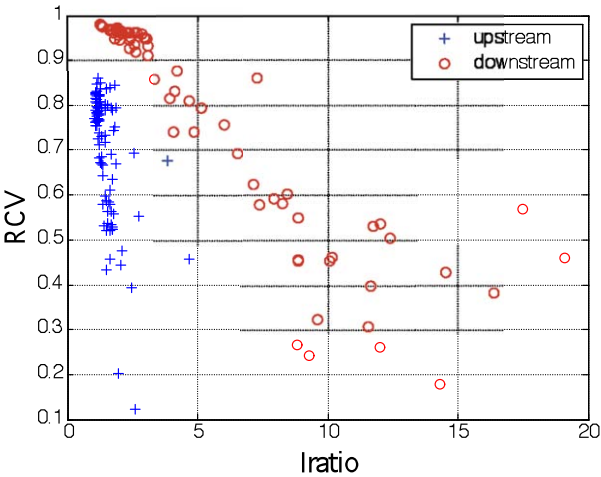


Fig. 4. I_{ratio} Vs RCV . The upstream sags are inside of area $I_{ratio} < 3$ and $RCV < 0,875$.

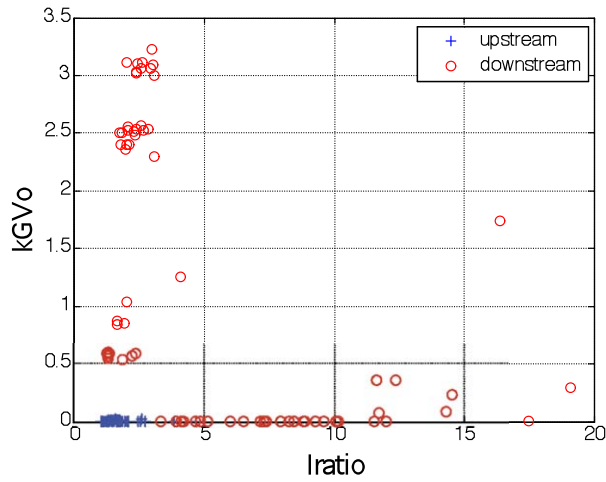


Fig. 5. I_{ratio} Vs GV_0 . The upstream sags are inside of area $I_{ratio}<3$ and $GV_0<100$.

Finally, the proposed rules for classification purposes are shown in **Table 5**. They are result of the combination between Rule 3 and Rule 4 with Z_{ratio} . It was added to the proposed rules because Z_{ratio} contains information about the current flow direction. Therefore, sag events caused by three-phase small motor starting connected in current flow direction (downstream) will be correctly classified.

These rules were tested; specificity and sensitivity equals to 0,93 and 0,00 were obtained respectively. Although the classification results are not better than Rule 3 and Rule 4 results, the proposed rules use the magnitude and current flow direction, which is very important in fault location analysis.

Table 5. Proposed Rules

| Proposed Rule |
|--|
| IF $RCV < Th_{RCV}$ AND $I_{ratio} = Th_{Iratio}$ AND $Z_{ratio} < Th_{Zratio}$ THEN upstream END |
| IF $GV_0 = Th_{GV_0}$ AND $I_{ratio} = Th_{Iratio}$ AND $Z_{ratio} < Th_{Zratio}$ THEN upstream END |

6. Conclusion

The problem of localizing the origin of voltage sag is an open problem. In this work, we have proposed the use of the CN2-SD subgroup discovery algorithm to perform a preliminary exploratory analysis to help in classifying the upstream/downstream voltage sag events.

Two extracted rules have been proposed to determine the upstream or downstream origin of voltage sag events. These determine the origin of sags taking into account the voltage magnitudes (RCV , GV_0), current flow direction (Z_{ratio}) and current magnitude

(I_{ratio}). Moreover, they can be implemented on power quality monitoring devices because the used features are simple to compute them.

As a future work, we should transfer the insights from subgroup discovery to other, more powerful methods to address the second part of the localization problem, that is, the accurate fault location. Moreover, the threshold selection process has to be automated.

References

- [1] Lavrac, N., Kavsec, B., Flach, P. & Todorovski, L. (2004). "Subgroup discovery with CN2-SD". *Journal of Machine Learning Research*, 5, 153-188.
- [2] Klösgen, W. "Explora: A Multipattern and Multistrategy Discovery Assistant. In Fayyad, U., Piatetsky-Shapiro, G., Smyth, P., & Uthurusamy, R. (Eds.), *Advances in Knowledge Discovery and Data Mining* (pp. 249-271). Menlo Park, California: AAAI Press, 1996.
- [3] Wrobel, S. "An algorithm for multi-relational discovery of subgroups". In *Proceedings of conference Principles Of Data Mining And Knowledge Discovery*, London, UK (pp. 78-87), 1997.
- [4] Klösgen, W., & Zytkow, J. "*Handbook of data mining and knowledge discovery*. New York: Oxford University Press, 2002.
- [5] C. Romero, P. González, S. Ventura, and others, "Evolutionary algorithms for subgroup discovery in e-learning: A practical application using Moodle data", *Expert Systems with Applications*, Elsevier Science.
- [6] C. Leborgne, D. Karlsson, J. Daalder, "Voltage Sag Source Location Methods Performance Under Symmetrical and Asymmetrical Fault Conditions", T&D Conference and Exposition: Latin America, IEEE/PES, pp: 1-6, August 2006.
- [7] A. Khosravi, J. Melendez, J. Colomer, "A Hybrid Method for Sag Source Location in Power Network", 9th International Conference. Electrical Power Quality and Utilisation. Barcelona, 9-11, October 2007.
- [8] Barrera, V; Berjaga, X; Meléndez, J; Herraiz, S and others, "Two methods for voltage sag source location", 13th International Conference on Harmonics & Quality of Power (ICHQP), Australia, 2008.
- [9] M. Bollen, "Algorithms for Characterizing Measured three-Phase Unbalanced Voltage Dips", *IEEE Transactions on Power Delivery*, vol. 18, no. 3, July 2003.
- [10] M. Bollen, D. Sabin, "International Coordination for Voltage Sag Indices", PES TD 2005/2006, May 21-24, 2006, pp.:229 – 234.
- [11] Kyoung K., Jin P., Jong L., Seon A., Seung M., "A Method to determine the Relative Location of Voltage Sag Source for PQ Diagnosis", ICEMS 2005, Proceedings of the Eighth International Conference on Volume 3, 27-29 Sept. 2005 pp.:2192 – 2197.
- [12] T. Tayjasanant, C. Li, and W. Xu, "A resistance sign-based method for voltage sag source detection," *IEEE TPWD*, vol. 20, pp. 2544-51, 2005.
- [13] A.K. Pradhan and A. Routray, "Applying distance relay for voltage sag source detection," *IEEE Transactions on Power Delivery*, vol. 20, pp. 529-31, 2005.
- [14] Dietterich, T, "Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms", *Neural Computation*, pp.: 1895-1924, 1998.

Statistical Monitoring of Injection Moulds

Xavier BERJAGA ^{a,b}, Joaquim MELENDEZ ^a and Alvaro PALLARES ^b

^a *Institut d'Informàtica i Aplicacions, Universitat de Girona,*

Av. Lluís Santalo. s/n, 17071, Girona, Spain

^b *Plastiasite S.A.*

Parc Tecnologic Valles, 08290, Barcelona, Spain

Abstract. In this paper a statistical based methodology to work over the principal component space on injection moulds is presented. The Multiway Principal Component Analysis is applied as a dimensionality reduction step, and fault detection assessment. This methodology allowed to analyse the behaviour of injections with the information directly received from sensors, what ended in a better process understanding. Results concluded that with a low number of variables (some principal components and two control statistics) is enough to detect abnormalities (too short injections, injections with low variability, etc.).

Keywords. Moulding process, Intelligent monitoring, Principal Component Analysis, Data Mining

Introduction

Injection moulding is one of the most important polymer processing operations in the plastic industry nowadays. Due to its ability to produce complex-shape plastic parts with good dimensional accuracy and very short cycle times, the injection moulding has become one of the processes that are greatly preferred in manufacturing industry [1].

The injection moulding process is a cyclic process that can be divided in four main parts: filling, packing, cooling and ejection. The filling stage consists in filling the mould with hot polymer. In the packing stage, new polymer melt is packed into the mould at a higher pressure in order to compensate the shrinkage. In the cooling stage the mould is cooled until its content is rigid enough to be ejected. Finally, in the ejection stage the mould is open, the part is ejected and closed another time, waiting for the beginning of the next cycle.

Two strategies can be followed to achieve the monitoring of moulding processes: the assessment of the moulding machine and the analysis of the moulding process. Referred to the first one, there are several topics of interest, such as determining the best strategy for the monitoring process [3], which are the effects of the variations of the parameters during the process [2], automatic selection of the best setting parameters [4], etc. On the other hand, the application of artificial intelligence and statistical techniques are used in the analysis of the moulding process. Neural Networks and Support Vector Machines [8] have been widely used to achieve pattern discovery goals [10] whereas statistical process control methods are being used for on line monitoring (SPC) [2].

This work is focused on the improvement of statistical monitoring strategies to determine the quality of the injection based on the the exploitation of redundancy captured

by sensors in the mould. Data mining principles will be applied in order to construct a statistical model of the Normal Operating Conditions (NOC) injections in the Principal Component Space. This model will characterise the injection procedure from the filling stage until the ejection of the manufactured product. Due to the large amount of data provided by sensors during this period of time and the necessity to preserve the most relevant information, the Multiway Principal Component Analysis (MPCA) will be used to reduce the dimensionality.

The paper is organised in 4 additional sections. In the next one, the Principal Components Analysis (PCA) basis are presented with its application in process monitoring. In Section 2 the MPCA methodology is presented, a PCA adaptation for monitoring finite duration (or batch) processes. And finally, in sections 3 and 4 a practical example of MPCA on a moulding process is explained.

1. PCA for multivariate process monitoring

In this section it will be presented the basis of PCA and how it can be applied for process monitoring and concretely for fault detection.

1.1. Introduction

Principal Component Analysis (PCA) is a technique for data compression and information extraction. PCA is used to find combinations of variables or factors that describe major trends in a data set [9].

Processes involving a large number of variable and affected of redundancy can be monitored using this technique. Observations during Normal Operation Conditions are used to build a data model. Further it is used to assess the behaviour of the process by checking new observations against this model in the principal component space (fault detection). In case of detecting an abnormal situation it is possible to identify (fault location, diagnosis) the variables, in the original space, responsible for that [5].

According to PCA basis the dataset, X (rows, n , represent observations -the injections- and the columns represent the variables), can be expressed as a linear combination of r new variables, assuming an error matrix E [7]:

$$X = \sum_{i=1}^r t_i \times p_i^T + E \quad (1)$$

Where t_i and p_i are named scores and loading vectors respectively and are computed to reflect relevant relation among variables (t_i) and observations (p_i). PCA assumes that the *loadings* associated to bigger eigenvalues represent the best directions for expressing the data upon based on the maximum variance criteria. According to this condition, those *loadings* associated to eigenvalues with a greater value, capture the majority of the variation. Retaining only few of them is possible to represent the majority of the variance of the original data set. Not retained information, represented by E (error, also known as residual matrix) is associated with noise and less meaningful variations. Thus, the first r principal components, instead the n original variables, build up a new space/model with a lower dimensionality than the original one. Projection of the data to this new space can be done using the following linear transformation:

$$T = XP \quad (2)$$

And the projection of scores, T , back onto the m -dimensional observation space can be computed with:

$$\hat{X} = TP^T \quad (3)$$

Where P is a matrix of r columns and n rows and is formed by $P = [p_1, p_2, \dots, p_r]$.

1.2. Process monitoring statistics in the principal component space

Two complementary control charts are usually used for multivariate process monitoring using PCA. The purpose is to assess new observations against the PCA model built during normal operation conditions. T^2 and Q statistics are used to build them. Control charts based on T^2 can be plotted based on the first r principal components [6]. For each observation the following statistic has to be computed:

$$T^2 = \sum_{j=1}^r \frac{t_j^2}{\lambda_j} \quad (4)$$

Where T^2 can be computed for each observation by adding the square of the r components t_j^2 weighted by their variances σ_j^2 (eigenvalues). This results in a measure of distance (Mahalanobis distance) of each observation to the center of the model. A graph, or control chart, built with this data is useful to detect variations in the plane of the principal components (r) greater than common-cause variations but preserving the structure gathered by the PCA model. Nevertheless, when a new event, x , in the process produces a large variation out of the hyperplane described by the r principal components this implies that the data structure has been broken. This type of event are detected by computing the Q -statistic or Squared Prediction Error (SPE) of the residual of each assessed observations defined as ([7]):

$$Q = (x - \hat{x})^T (x - \hat{x}) \quad (5)$$

Where \hat{x} is the reconstruction of the observation computed using the loading matrix, P , as it is shown for the original data set, X , in Equation 3. Q -statistic is much more sensitive than T^2 to changes in the process structure. This is because Q during normal operation conditions is very small (typically associated to noise) and therefore any minor change in the process will affect the correlation structure of observed data. T^2 represents a greater variance and therefore it is less sensible to small variations in the process (Figure 1).

2. Multi-way Principal Component Analysis (MPCA)

The PCA methodology presented before can be directly applied on two dimensional matrices (*observations* \times *variables*). Finite duration processes are usually represented by time series of variables representing the execution of the process. Consequently, a three dimensional matrix is needed to represent the data set (*observations* \times *variables* \times

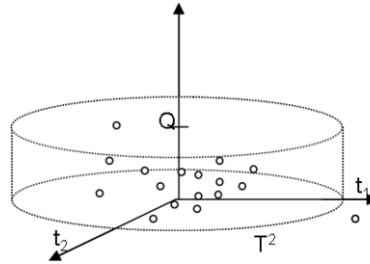


Figure 1. Graphical representation of Q and T^2 statistic

time). This added complexity implies to perform a two steps preprocessing before applying the PCA methodology: unfolding and scaling. Unfolding consists in rearranging the 3D matrix into a 2D one, whereas scaling is the procedure applied to guarantee the same importance to any variable.

2.1. Unfolding the data

From the six feasible unfolding directions, only 2 of them are meaningful for monitoring: unfold in the observation (injection) direction (batch-wise) and unfold in the variable direction. When monitoring is performed off line (once the process injection is finished) the best unfolding direction is the batch-wise (injections are kept as rows) [7]. The resulting data matrix is shown in Figure 2 b), where variables \times time represent the sequence of samples retained for each variable.

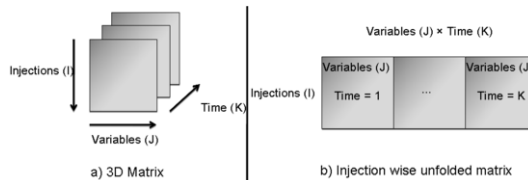


Figure 2. Process wise unfolding of the original 3D matrix

2.2. Auto scaling the data

As can be seen in Figure 3 a), some variables after the unfolding present different range of variation. In order to have all variable having the same importance, the data has to be normalised. From the several possible approaches [7], the one used here is the so-called auto scaling, which formulation is as follows:

$$X_n = \frac{X_u - \bar{x}}{\sigma} \quad (6)$$

Where X_n is the new normalised value of the unfolded data, X_u stands for the unfolded data, \bar{x} is referred to the vector of means of each variable at each time instant

and σ is the vector of standard deviations of each variable at each time instant. After applying this normalisation step, the resulting process is presented in Figure 3 b).

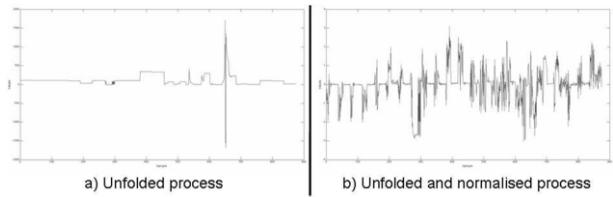


Figure 3. Effect of normalising an unfolded process

3. MPCA model creation

Once the required steps for applying the basic PCA methodology on three dimensional matrices have been explained in the previous section, in the next sections (Section 3 and Section 4) it will be applied the MPCA methodology to analyse an injection mould process. First of all, the data pre-treatment phases done over the original data set (divide the data in injections, variable filtering) will be exposed. Finally, the procedure to select the number of components selected to create the model (since the monitoring statistics are sensitive to it).

3.1. Building the 3D Matrix

The original data set was formatted as a two dimensional matrix where rows represent all time instants registered in the file (multiple injections) and columns represent all measured variables. In order to obtain a model for the injections it is needed to identify the beginning and end of each injection and build the 3-D matrix referred in the previous section. Special interest in this step is because the need of data perfectly aligned (correspondence among instances of different injections) when building the 3-D matrix. One of the most critical variables of the process, temperature sprew, represented by X_{16} , has been used to identify the beginning of each injection and the duration. The resulting division of the original data is presented in Figure 4.

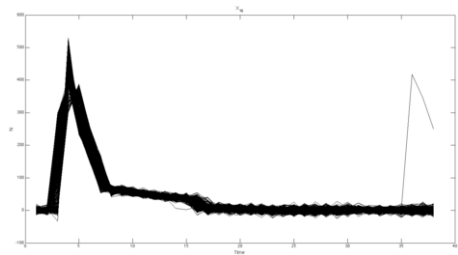


Figure 4. Division of the data in injections using X_{16}

3.2. Variable filtering

Due to that not all variables presented an important variability, those controlled variables (temperatures) were erased after verifying their low information gain when building the statistical model. The remaining variables that will be used in the MPCA methodology can be seen in Figure 5.

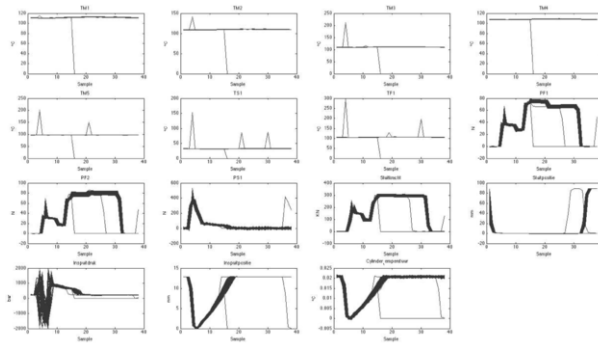


Figure 5. Variable retained for the MPCA model creation

3.3. Selection of the number of principal components

One of the most important points in the MPCA model generation is the number of principal components to retain, since Hotelling's T^2 and Q-statistic are sensitive to it. Several criteria for selecting the number of components have been applied (Scree plot test, $\lambda > 1$, representativity of original variables). Since results offered by those methods were very different in terms of number of principal components, finally the minimum given by all of them has been selected.

4. MPCA model exploitation

The model generated has to be analysed looking for cases too far from the centre. When those cases are removed, the model used for fault detection is defined. Since the resulting model is more refined, those cases will be detected as faults. In Section 4.1 the criteria used for finding those covered faults is explained. Those injections will be studied in Section 4.2 in order to find some common points among them. Finally, some conclusions and future work will be explained.

4.1. Fault detection using the MPCA model

Before building the final model for monitoring, injections too far from the model centre are removed to get a more consistent model. Once those points are projected to the new model, they will be classified as faults. The square points of Figure 6 show those injections that have been removed because they were very far from the center of the cloud

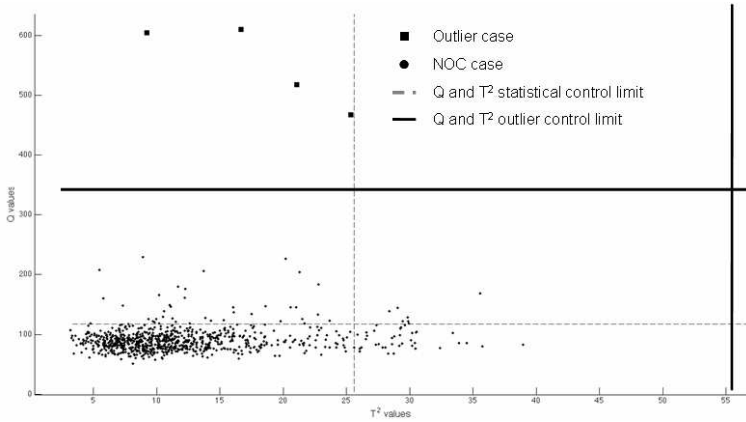


Figure 6. Projection over the Q - T^2 space of the injections

of points (injections), and will be studied in the next section, and where detected using Equation 7.

$$FAULT(c_i) = \{T_{c_i}^2 \geq \theta_{T^2} \vee Q_{c_i} \geq \theta_Q\} \quad (7)$$

Where θ_{T^2} and θ_Q are respectively the T^2 and Q -statistic fault thresholds.

4.2. Faulty injections analysis

All those cases that were considered to be faulty according to Equation 7 were analysed, resulting in those points:

- 2 of the outliers were those cases that presented a period value lesser than the mode of values (38). One presented an extremely large value of T^2 (resulting in a modification of the first direction of variance in the PCA, and the other presented the increase in the Q -statistic.
- The remaining erased cases were the cause of misalignments on the variables that were not used neither for aligning nor dividing.
- When comparing the nearest point to the centre of the model, it was found that some injections that were considered outliers presented less variation than the reference injection in the lowest variation variables.

4.3. Conclusions and future work

In this paper, a procedure for monitoring of injection moulds using data mining principles and MPCA model exploitation has been presented. Taking into account that no information was known about the model in question, results allowed the identification of those cases that can be considered in the NOC region, and those that were suspicious of presenting some abnormalities. This identification is done from the analysis of the captured signals of the process sensors, what means that an explicit knowledge of the process to analyse is not needed. As a complement of the fault detection conducted in this work, contribution analysis whenever a fault is found should be analysed. Related

to the model refinement, it should be tested with other statistical criteria in order to corroborate its accuracy and reliability. Another interesting procedure would be the usage of Case-Based Reasoning for the fault diagnosis, and simplifying the process of casualty assignment by a similarity criterion.

Acknowledgements

This research has been supported by EMOLD project (COLL-CT-2006-030339). The authors would like to thank TNO for delivering the data this work is based on. The authors also would like to recognise Magda Ruiz from Universitat de Girona and Alberto Ferrer from Universidad Politécnica de Valencia for their ideas and assessment in the MPCA model creation and exploitation. We also greet Encarna Escudero and Francesco Puliga from ASCAMM Foundation for their help on understanding the basis of injection moulding processes and their principal points to study.

References

- [1] A. T. Bozdana and O. Eyercioglu. Development of an expert system for the determination of injection moulding parameters of thermoplastic materials: Ex-pimm. *Journal of Materials Processing Technology*, 128:113–122, 2002.
- [2] J. Cao, Y. S. Wong, and K. S. Lee. Application of statistical process control in injection mould manufacturing. *International Journal of Computer Integrated Manufacturing*, 20(5):436–451, 2007.
- [3] C. Collins. Monitoring cavity pressure perfects injection molding. *Assembly Automation*, 19(3):197–202, 1999.
- [4] X. Jin and X. Zhu. Process parameters' setting using case-based and fuzzy reasoning for injection molding. proceedings. *3rd World Congress on Intelligent Control and Automation*, pages 335–340, 2000.
- [5] R. A. Johnson and D. W. Wichern. *Applied Multiway Statistical Analysis*. Englewood Cliffs, Prentice-Hall International, 1992.
- [6] J. F. MacGregor. Multivariate statistical approaches to fault detection and isolation. *5th IFAC SAFE-PROCESS*, 2003.
- [7] P. Nomikos and J. F. MacGregor. Monitoring batch processes using multiway principal component analysis. *AIChE*, 40(3):1361–1375, 1994.
- [8] B. Ribeiro. Support vector machines for quality monitoring in a plastic injection molding process. *IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews*, 35(3):401–410, 2005.
- [9] B. M. Wise, N. B. Gallagher, S. Watts, D. D. White JR, and G. G. Barna. A comparison of pca, multiway pca, trilinear decomposition and parallel factor analysis for fault detection in a semiconductor etch process. *Journal of Chemometrics*, 13:379–396, 1999.
- [10] S. L. B. Woll and D. J. Cooper. On-line pattern-based part quality monitoring of the injection molding process. *Polymer Engineering and Science*, 36(11):1477–1488, 1996.

On the Dimensions of Data Complexity through Synthetic Data Sets

Núria MACIÀ, Ester BERNADÓ-MANSILLA, and Albert ORRIOLS-PUIG

Grup de Recerca en Sistemes Intel·ligents

Enginyeria i Arquitectura La Salle - Universitat Ramon Llull

Quatre Camins 2, 08022, Barcelona (Spain)

{nmacia,esterb,aorriols}@salle.url.edu

Abstract. This paper deals with the characterization of data complexity and the relationship with the classification accuracy. We study three dimensions of data complexity: the length of the class boundary, the number of features, and the number of instances of the data set. We find that the length of the class boundary is the most relevant dimension of complexity, since it can be used as an estimate of the maximum achievable accuracy rate of a classifier. The number of attributes and the number of instances do not affect classifier accuracy by themselves, if the boundary length is kept constant. The study emphasizes the use of measures revealing the intrinsic structure of data and recommends their use to extract conclusions on classifier behavior and their relative performance in multiple comparison experiments.

Keywords. Data complexity, Classification, Dimensionality, Synthetic data sets

Introduction

The analysis of data complexity [3] is an emergent research area that studies the characterization of data sets to understand their intrinsic structure and to identify to what extent useful patterns can be extracted from them. Recent investigations have characterized the complexity of data sets by a set of measures describing the geometry of classes around the feature space and have found correlations with the error of classifiers [8]. Also, preliminary studies [4] have tried to identify categories of classification problems according to complexity and relate optimal classifiers to each group. These kinds of studies may enhance our current understanding of classifier behavior such as its expected accuracy for a given problem and its comparative advantages with respect to other methods in certain types of domains.

Simultaneously with this line of study, there are many investigations trying to propose new classification algorithms or improve the existing ones. Such investigations are usually supported by experimental validation of the method of interest across a selection of several real-world data sets, and possibly comparing the given method with others to highlight its advantages. In these experiments, there are few cases where the given classifier outperforms the other(s) in all the domains. Instead, the method generally provides better performance in some problems and worse performance in others. A mandatory

conclusion of this approach is to identify in which cases the given method will be best and worst. Usually, there is no such understanding. Then, the desired result is that the method performs best more times than worst. This also has an important limitation because we cannot generalize whether the classifier will still behave best in a different set of problems, or whether a new problem will be properly classified by the given algorithm. Furthermore, given a new problem, one can never know whether there is a classifier that will perform better or whether we have already reached the maximum accuracy bound.

A classic investigation of this type involves a table of accuracy rates of different classifiers across different real-world problems (usually from public repositories). These data sets are barely characterized by the number of classes and the dimensionality. We aim to discuss why we cannot find correlations between classifier behavior and data set given this characterization. We study the influence of dimensionality on data complexity and see whether this can be somehow related with classifier performance. To perform this study, we design a set of synthetic data sets that allow us to vary the data dimensionality while maintaining a given class structure. We demonstrate that data dimensionality does not affect classifier error by itself. In summary, we emphasize the use of complexity measures that look at the intrinsic structure of the data set rather than the external appearance.

The remainder of the paper is organized as follows. Section 1 shows an example of a typical experimentation where no informational value can be extracted by relating classifier behavior to the external characterization of the data set. Section 2 briefly reviews the analysis of data complexity and describes one of the most important measures describing the inherent structure of data. Section 3 introduces this measure and finds correlations with classifier error. Then, we study the influence of data dimensionality on data complexity and classifier behavior. Finally, Section 4 presents the conclusions and future work.

1. Motivation

Many studies claim the universality of a given classification method by testing it over an apparent large variety of problems. Table 1 shows an example of a typical experimentation framework, where the accuracies of several classifiers are compared on a set of problems extracted from the UCI repository [2]. Each problem is characterized by the number of classes (not depicted here, since we restricted the study to binary class problems), the number of attributes, and the number of instances. These two latter dimensions are frequently misunderstood as indicators of the problem complexity. Often one assumes that the higher the dimensionality, the higher the complexity. Certainly, these dimensions provide an estimation of the data volume and can have some relationship with data sparsity. However, there are other complexities hidden in the data sets that may be more influential. Then, we aim to study the following issue: what is the relationship, if any, between these dimensions and classifier performance?

To answer this question, we first carried out an experiment with an enhanced set containing 264 problems, which were obtained from a selection of 54 problems from the UCI repository transformed into binary class problems. Figures 1(a) and 1(b) depict the relation between the accuracy of several classifiers—an induction tree (C4.5) [10], an instance based learning IB3 [1], a rule learner (PART) [11], and a support vector ma-

Table 1. Classical experimental framework

| Data set | #Attr | #Inst | C4.5 | IB3 | PART | SMO | B |
|-------------------------------|-------|-------|--------|--------|--------|--------|--------|
| Abalone | 8 | 4177 | 0.9998 | 0.9998 | 0.9998 | 0.9998 | 0.0005 |
| Balance Scale | 4 | 625 | 0.8467 | 0.8866 | 0.8625 | 0.9297 | 0.2240 |
| Breast Cancer Wisconsin | 30 | 569 | 0.9367 | 0.9719 | 0.9332 | 0.9771 | 0.0721 |
| Chess (King-Rook vs. King) | 6 | 28056 | 0.9791 | 0.9010 | 0.9975 | 0.9003 | 0.1642 |
| Glass Identification | 9 | 214 | 0.8071 | 0.8355 | 0.8407 | 0.7104 | 0.3224 |
| Heart Disease | 13 | 303 | 0.8037 | 0.7963 | 0.7444 | 0.8407 | 0.3667 |
| Hepatitis | 19 | 155 | 0.8008 | 0.7996 | 0.8020 | 0.8804 | 0.2839 |
| Ionosphere | 34 | 351 | 0.9120 | 0.8518 | 0.9118 | 0.8776 | 0.2308 |
| Iris | 4 | 150 | 0.9933 | 1 | 0.9933 | 1 | 0.0133 |
| Lenses | 4 | 24 | 0.8167 | 0.9333 | 0.8833 | 0.8000 | 0.1250 |
| Letter Recognition | 16 | 20000 | 0.9960 | 0.9994 | 0.9965 | 0.9916 | 0.0016 |
| Lung Cancer | 56 | 32 | 0.7583 | 0.7333 | 0.7583 | 0.6500 | 0.5000 |
| Optical Recognition | 64 | 5620 | 0.9939 | 0.9996 | 0.9957 | 0.9977 | 0.0007 |
| Pima Indians Diabetes | 8 | 768 | 0.7566 | 0.7382 | 0.7161 | 0.7669 | 0.4375 |
| Statlog (Image Segmentation) | 19 | 2310 | 0.9939 | 0.9952 | 0.9931 | 0.9965 | 0.0010 |
| Statlog (Vehicle Silhouettes) | 18 | 846 | 0.7695 | 0.7731 | 0.7766 | 0.7494 | 0.3652 |
| Thyroid Disease | 5 | 215 | 0.9344 | 0.9485 | 0.9299 | 0.7907 | 0.1023 |
| Waveform Database Generator | 21 | 5000 | 0.8290 | 0.8496 | 0.8360 | 0.8588 | 0.2384 |
| Wine | 13 | 178 | 0.9604 | 0.9833 | 0.9604 | 0.9889 | 0.0674 |
| Yeast | 8 | 1484 | 0.7223 | 0.7069 | 0.7156 | 0.6880 | 0.4501 |

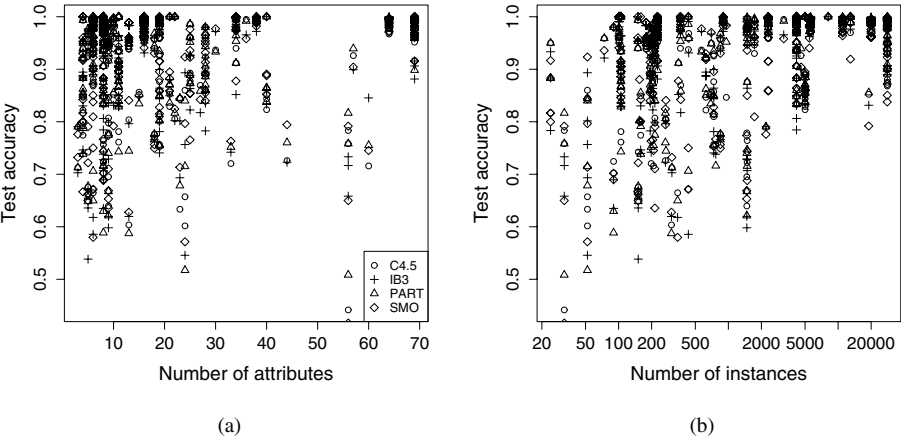


Figure 1. Accuracy of classifiers with respect to (a) number of attributes and (b) number of instances in logarithmic scale

chine (SMO) [9]– and the number of instances and number of attributes of the data sets respectively. From these plots, we cannot observe any pattern nor any sort of correlation between the classifiers’ accuracy and these data set characteristics. Thus, it seems that it is necessary to find out other descriptors to characterize data and see if we can find better estimators of data set complexity. The best characterization would be the one able to provide a predictive estimation of learner behavior.

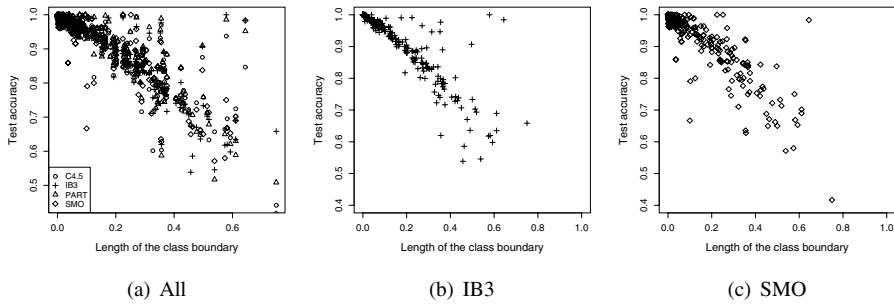


Figure 2. Accuracy of classifiers with respect to the length of the class boundary

2. Data Complexity

The analysis of data set complexity provides a general framework to characterize data sets and find explanations about classifier behavior. Previous studies by Ho and Basu [8] proposed a set of measures that described different aspects of complexity: a) the discriminative power of attributes, b) the separability of classes, c) the geometry of manifolds expanded by each of the classes, and d) the sparsity. These descriptors were found useful to estimate classifier performance [5]. Also, data set characterization was used to investigate the domains of competence of classifiers [4] and study the suitability of classifier ensembles [7]. In all these cases, measures of class separability were identified as the most relevant for the characterization of data set complexity. In particular, the *length of the class boundary* achieved high correlations with several algorithms' performance and thus, was identified as a good estimator of classifier error.

The length of the class boundary counts the proportion of points in the data set that lie near the class boundary. It is computed as follows. Given a data set, a *minimum spanning tree* (MST) is built with all the points of the data set, according to their Euclidean distances. Then, the number of edges connecting points of opposite classes is counted and divided by the total number of connections. This ratio is taken as the measure of boundary length. If the data set is highly interleaved, which means that points of opposite classes are very close to each other, the measure will be high (close to 1). Otherwise, if classes are well separated, the boundary length will be small.

We computed this measure to the 264 data sets to analyze whether it could provide better insights of classifier behavior than with measures based on the dimensionality of the data sets. The obtained results are depicted in figure 2. Plot 2(a) shows all the results of the different classifiers, plot 2(b) shows the most correlated classifier, which is IB3, and plot 2(c) the least correlated classifier, which corresponds to SMO. Even in the worst case, the boundary length presents almost a linear correlation with the accuracy of the learners. This means that the length of the class boundary could be an interesting data descriptor whose information could be used to predict classifier performance. Therefore, we considered the length of the class boundary as the main responsible for data complexity and studied how the data set dimensionality could alter classifier response to such complexity. The next section explains the procedure designed to perform this study.

3. Experimentation and Results

The previous section demonstrated that the boundary length is a good estimator of data complexity for several types of classifiers. However, we wonder whether the length of the class boundary is the only responsible for complexity. One of the relevant difficulties that classifiers may encounter is the sparsity of the training data set, i.e., the lack of representative examples. Also, a high number of attributes may hinder the performance of many classifiers. In general, the researcher knows that a low ratio between the number of instances and the number of features usually denotes sparsity and thus, a complex problem. However, this is not necessarily related to classifier accuracy as figure 1 demonstrated.

To perform such study, we could not rely on real-world data sets because we could not control the real complexity of the data set. We instead designed a set of synthetic problems that allowed us to vary these three dimensions of complexity (boundary length, number of attributes, and number of instances) independently. The data sets were built to study how a varying number of attributes and a varying number of instances influence classifier accuracy for a given fixed boundary length. For this purpose, the data sets were built according to the following procedure.

Each data set is characterized by the number of features m , the number of instances n , and desired boundary length b . Once these parameters were set, we generated n points with the values of the attributes distributed uniformly in the m -dimensional feature space. Then, we constructed the MST connecting the points according to Euclidean distances. To achieve the desired boundary length b , we label the classes of the points until there are $b \cdot (n - 1)$ edges connecting points of different classes. In fact, for a given MST and boundary length b , there are $2 \cdot \binom{n-1}{(n-1)-p}$ different labelings, where p is the number of edges joining different classes, i.e., $p = b \cdot (n - 1)$. Among all these possible labelings, we perform an heuristic search that obtains a labeling that does not incurs in class imbalances. The presence of class imbalances could also be a factor of complexity to some classifiers, so we did not aim to be disturbed by this possible complexity in our analysis. Future work will include this issue to further understand the complexity of data.

To analyze classifier behavior to increasing number of attributes, we prepared a collection of 2000 artificial data sets. We chose medium boundary length $b = 0.3$ and $b = 0.4$, with 1000 data sets each. For a given boundary length, we built data sets with number of attributes $m = \{2, 4, 6, 8, 10, 15, 20, 25, 50, 100\}$ and a fixed number of instances $n = 200$. For each value m , there were 100 data sets, where each data set contained a different distribution of points in the feature space and thus a different MST which was then labeled to achieve the given boundary length b . Thus, we ranged from a ratio instances/attributes of $200/2$ to a ratio of $200/100$. Note that the latter represents a case where one would usually identify a difficult problem.

As the boundary length is based on the distance computation, we selected three classifiers that do not use distances in their learning process to avoid correlations with the metric. We chose three classifiers belonging to different learning paradigms: SMO, C4.5, and PART. We used a 10-fold cross-validation procedure [6] to estimate the classifier's accuracy with each data set. The algorithms were run with the software Weka [11] with their default configuration.

Figure 3 summarizes the results of the three classifiers, where the x-axis of each graph plots the number of attributes and the y-axis the accuracy rate. For a given number

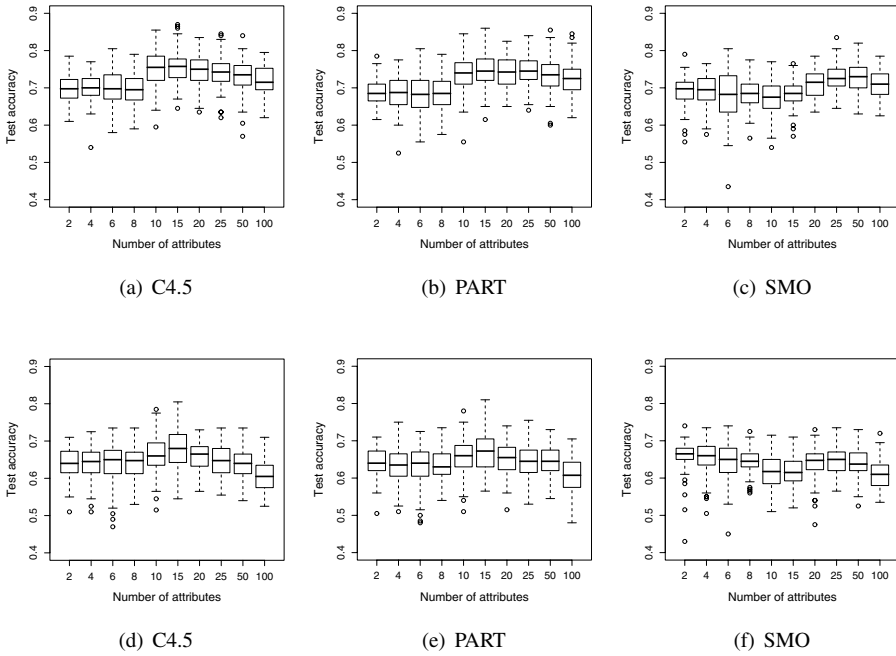


Figure 3. Accuracy rate of different classifiers with a fixed boundary length and increasing number of attributes

of attributes m , the corresponding boxplot represents the range of values of the accuracy rate obtained by the classifier in the 100 data sets. The three upper plots refer to boundary complexity $b = 0.3$ and the lower ones to boundary $b = 0.4$. Note that the three classifiers present similar behavior within each boundary complexity. The interquartile range of the boxplots, which contain 50% of the results, gives accuracy rates ranging in the interval $[0.6, 0.8]$ for boundary length $b = 0.3$ and $[0.62, 0.7]$ for $b = 0.4$. The median of accuracy rates gives a value close to $1 - b$. Taking into account the whole spread of values in the boxplot, there is a fairly high dispersion of accuracy rates with respect to a given boundary length b . This indicates that there are other complexity issues affecting classifier error, which cannot be accounted only by the boundary length. Nevertheless, it seems that this complexity is not due to the increasing number of attributes, since in all cases, the median and spread of the accuracy rates are very similar. These results indicate that the number of attributes is not a complexity dimension influencing classifier error by itself. That is to say, the number of attributes can be a factor of complexity that can be modeled by the boundary metric. An increasing number of attributes does not necessarily incur in higher errors unless this alters the measure of boundary length. This is an interesting hypothesis that should be further analyzed but it supports the fact that classifier error does not correlate necessarily with the number of attributes as shown in figure 1(a).

In the second part of the analysis, we studied the variability of the accuracy rate with respect to the number of instances. As before, we generated 1000 artificial problems for $b = 0.3$ and $b = 0.4$. We fixed the number of attributes $m = 2$, and built data sets with the

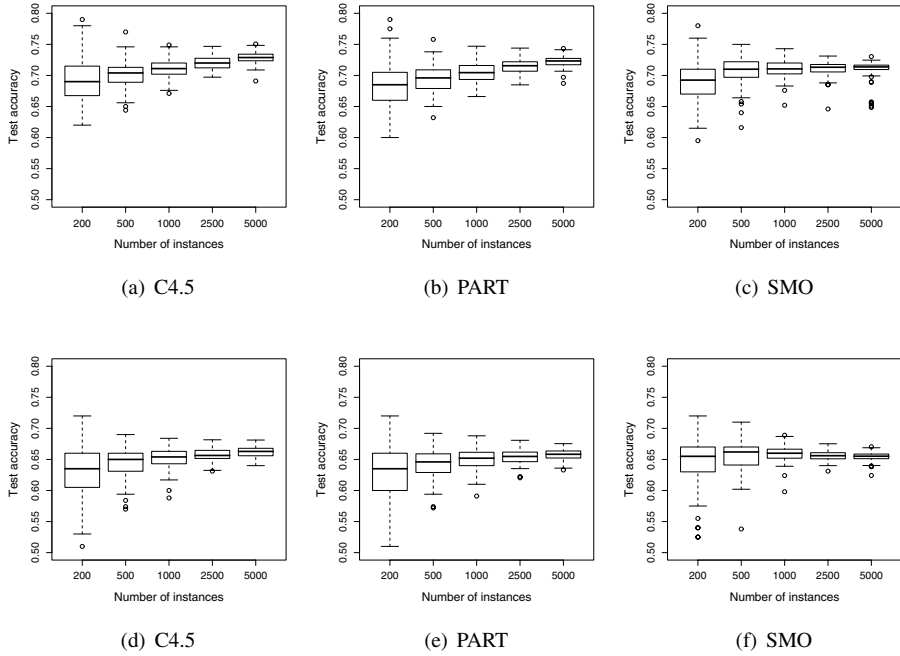


Figure 4. Accuracy rate of different classifiers with a fixed boundary length and increasing number of instances

following number of instances $n = \{200, 500, 1000, 2500, 5000\}$. Thus, for each boundary length b and instances n , we generated 1000 data sets. The ratio instances/features ranged from $200/2$ to $5000/2$.

Figure 4 depicts the classification accuracy obtained by the classifiers. The x-axis refers to the number of instances and the y-axis the accuracy rate. For a given number of instances n , a boxplot represents the range of values of the accuracy rate obtained by the classifier in the 100 data sets. The three upper plots refer to boundary complexity $b = 0.3$ and the lower ones to boundary $b = 0.4$. Note that an increasing number of instances allows for slightly better accuracy rates in general. This happens for all the three classifiers and is more notorious with boundary length $b = 0.3$. However, this increase of accuracy rate is still within the spread of the boxplots with the fewest number of instances. The most relevant observation is that the spread of accuracy rates decreases with increasing number of instances. A higher number of instances allows for further redundancy, which probably means that the classifiers can generalize better. Again, observe that the three classifiers have a very similar behavior. Even though they represent three different learning schemes, their accuracy rates have the same tendency. There are no significant differences among the classifiers. Our interpretation of this fact is that the complexity of the data set is more influential to the classifier's behavior than the bias of the classifier itself. This is true for the current set of synthetic problems and may not be extrapolated to real-world problems. However, this reminds us that much caution should be taken when we extract conclusions from the comparisons of several classifiers on a small set of real world problems. Under a few number of data sets, the results of a particular classifier

could be easily biased by the particular selection. We should investigate whether the general tendency of the classifier is the same as other classifiers on a larger set of problems or on the contrary there are significant differences. We could also enrich the study by the introduction of complexity measures such as the length of the class boundary to fully understand the behavior of the classifiers. If significant differences are identified, the use of complexity characterization can also be useful to detect when these differences occur.

4. Conclusions

We analyzed learner behavior through a data characterization based on three dimensions: the number of attributes, the number of instances, and the length of the class boundary. Empirically, we showed that the number of attributes and the number of instances are not correlated to the classifiers' accuracy. Although this lack of correlation was already known, researchers still use these measures to characterize the data sets. We found that the boundary length can be considered a significant factor to assess the complexity and estimate classifier accuracy. In this study, we realized by means of synthetic data sets that the dimensionality does not affect classifier accuracy and the information on complexity provided by the number of attributes and instances can be embedded in the measure of length of the class boundary. Nevertheless, the variability observed in the results indicates that other complexities may be involved in data characterization. As a future work we aim to extend this analysis to other complexity measures.

This paper highlights the benefits of using synthetic data sets along the experiments because they allow us to vary these three dimensions independently and work under a controlled scenario. However, the generated synthetic data sets do not contain a real structure concerning the distribution of points in the feature space since the points follow a uniform distribution. Because of this distribution, we are modeling an upper bound of complexity. We expect real-world problems to have more structure due to some underlying physical process and thus, points can be grouped in more easily identifiable patterns or clusters. The synthetic data sets could be also designed to be closer to such real-world problems.

Another interesting observation from the study is that the classifiers behaved similarly. In any case, the dependence between the problem's structure and the classifier's behavior could mean that the classifier's error is mainly due to the difficulty of the problem rather than the classifiers' own constraints. In fact, we have already attained a mature development of the classifiers and we must go one step further and tackle data complexity analysis. If we can identify the real structure of the problem and have some control over the sampling processes, we could design algorithms that reduce the complexity of the problem.

Acknowledgements

The authors would like to thank *Enginyeria i Arquitectura La Salle, Universitat Ramon Llull*, the *Ministerio de Educación y Ciencia* for its support under project TIN2005-08386-C05-04, *Generalitat de Catalunya* for its support under grants 2005FI-00252 and 2005SGR-00302, and the *Govern d'Andorra* for its research grant.

References

- [1] D. Aha, D. Kibler, and M. Albert. Instance-based learning algorithms. *Machine Learning*, 6(1):37–66, 1991.
- [2] A. Asuncion and D. Newman. UCI machine learning repository. University of California, Irvine, School of Information and Computer Sciences, 2007. <http://www.ics.uci.edu/~mllearn/MLRepository.html>.
- [3] M. Basu and T. K. Ho. *Data Complexity in Pattern Recognition*. Springer-Verlag New York, 2006.
- [4] E. Bernadó-Mansilla and T. K. Ho. On classifier domains of competence. In *17th International Conference on Pattern Recognition*, volume 1, pages 136–139. IEEE Computer Society, 2004.
- [5] E. Bernadó-Mansilla, T. K. Ho, and A. Orriols-Puig. Data complexity and evolutionary learning. In *Data Complexity in Pattern Recognition*, pages 115–134. Springer, 2006.
- [6] T. Dietterich. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1924, 1998.
- [7] T. K. Ho. Data complexity analysis for classifier combination. In *MCS '01: Proceedings of the Second International Workshop on Multiple Classifier Systems*, pages 53–67, London, UK, 2001. Springer-Verlag.
- [8] T. K. Ho and M. Basu. Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):289–300, 2002.
- [9] J. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods: Support Vector Learning*, pages 185–208, Cambridge, MA, USA, 1998. MIT Press.
- [10] J. Quinlan. *C4.5: Programs for machine learning*. Morgan Kaufmann Publishers, San Mateo, California, 1993.
- [11] I. H. Witten and E. Frank. *Data Mining: Practical machine learning tools and techniques*. Morgan Kaufmann Publishers, 2nd edition, 2005.

Can Evolution Strategies Improve Learning Guidance in XCS? Design and Comparison with Genetic Algorithms based XCS

Sergio MORALES-ORTIGOSA, Albert ORRIOLS-PUIG, and
Ester BERNADÓ-MANSILLA

*Grup de Recerca en Sistemes Intel·ligents
Enginyeria i Arquitectura La Salle, Universitat Ramon Llull
Quatre Camins 2, 08022, Barcelona (Spain)*

Abstract. XCS is a complex machine learning technique that combines credit apportionment techniques for rule evaluation with genetic algorithms for rule discovery to evolve a distributed set of sub-solutions online. Recent research on XCS has mainly focused on achieving a better understanding of the reinforcement component, yielding several improvements to the architecture. Nonetheless, studies on the rule discovery component of the system are scarce. In this paper, we experimentally study the discovery component of XCS, which is guided by a steady-state genetic algorithm. We design a new procedure based on evolution strategies and adapt it to the system. Then, we compare in detail XCS with both genetic algorithms and evolution strategies on a large collection of real-life problems, analyzing in detail the interaction of the different genetic operators and their contribution in the search for better rules. The overall analysis shows the competitiveness of the new XCS based on evolution strategies and increases our understanding of the behavior of the different genetic operators in XCS.

Keywords. Evolutionary Algorithms, Genetic Algorithms, Evolution Strategies, Learning Classifier Systems, Supervised Learning, Data Mining.

Introduction

Learning Classifier Systems (LCSs) [8] are machine learning techniques that learn *rule sets* on-line through the interaction with an *environment* that represents a *stream of labeled examples*. In the recent years, XCS [13], the most influential LCS, has arisen as a promising technique for classification tasks and data mining, showing its competitiveness with respect to highly-used machine learning techniques such as the decision tree C4.5 and support vector machines [10]. XCS consists of a complex architecture that evolves a rule set by means of the interaction of two main components: (i) a rule evaluation system and (ii) a rule discovery procedure. The rule evaluation system, based on credit apportionment techniques, is responsible for evaluating the quality of the rules on-line with the information provided by the environment. This component has received an in-

creasing amount of attention during the last few years, resulting in several improvements that enabled the system to solve problems that previously eluded solution [3]. The rule discovery procedure, driven by a genetic algorithm (GA), is responsible for providing new promising rules to the system. Whereas the evaluation component has been studied in detail, the discovery component has received little attention, remaining practically unchanged from its initial conception.

In this paper, we aim at analyzing the behavior of the discovery component in XCS. For this purpose, we design a new discovery procedure based on evolution strategies (ESs) [12] to drive the discovery of new rules in the on-line learning architecture. We modify the interval-based rule representation of XCS [14] by introducing a vector of strategy parameters and adapt the selection, recombination, and mutation operators of ESs to let them deal with interval-based rules. Both original XCS and XCS based on evolution strategies are compared on a collection of real-life problems. The interaction of different genetic operators, i.e., the combination of selection and mutation and the combination of selection, crossover, and mutation, is carefully studied for each one of the two systems. This study not only shows the performance improvements due to the new discovery component, but also enables us to increase our comprehension of how the evolutionary genetic process works.

The remainder of this paper is structured as follows. Section 1 gives a brief description of XCS. Section 2 presents the new ES-based discovery component introduced to XCS, detailing the modifications introduced to XCS representation and to the different genetic operators. Section 3 provides the methodology followed in the experimentation and Section 4 analyzes the obtained results, especially highlighting the interaction of different genetic operators. Finally, Section 5 summarizes, concludes, and presents some future work lines that will be followed in light of the results presented in the current study.

1. XCS in a Nutshell

As follows, we provide a brief description of XCS focused on classification tasks. For an algorithmic description the reader is referred to [5].

1.1. Knowledge Representation

XCS evolves a population [P] of classifiers. Each classifier consists of a production rule and a set of parameters. The production rule takes the following form: **if** $x_1 \in [l_1, u_1] \wedge \dots \wedge x_\ell \in [l_\ell, u_\ell]$ **then** *class*. That is, each variable of the condition x is represented by an interval $[l_i, u_i]^\ell$ (ℓ is the number of input attributes of the problem). Then, a rule matches an input instance $e = (e_1, e_2, \dots, e_\ell)$ if $\forall_i l_i \leq e_i \leq u_i$.

Each classifier has three main parameters: (i) the payoff prediction p , an estimate of the reward that the system will receive if the class of the rule is selected as output, (ii) the prediction error ϵ , which estimates the error of the payoff prediction, and (iii) the fitness F , which is computed as an inverse function of the prediction error.

1.2. Process Organization

XCS learns the data model on-line by means of the interaction with an environment which represents a stream of examples. That is, at each learning iteration, XCS receives a new training example $e = (e_1, e_2, \dots, e_\ell)$ and the system creates a *match set* [M], which consist of all the classifiers in [P] whose condition matches e . If any of the classes is not represented in [M], the covering operator is triggered, creating a new classifier that predicts the missing class and whose condition is generalized from the input as $\forall_i^\ell : l_i = e_i - rand(0, r_0)$ and $u_i = e_i + rand(0, r_0)$, where r_0 ($0 < r_0 < 1$) is a configuration parameter that determines the initial generalization degree. The next step depends on whether the system is in exploration (training) mode or in exploitation (test) mode. In exploration mode, the system randomly chooses one of the possible classes and builds the action set [A] with all the classifiers in [M] that advocate the selected class. The action set represents the niche of similar classifiers where both the parameter update procedure and the genetic algorithm take place. The parameters of all the classifiers in [A] are updated according to a generalized version of Q-learning (see [13]). The genetic algorithm is applied as explained in the next subsection. In exploitation mode, the classifiers in [M] vote, according to their fitness, for the class they predict. The most voted class is selected as output.

1.3. Discovery Component

XCS applies a steady-state niche genetic algorithm (GA) [7] to discover new promising rules. The GA is triggered on [A] when the average time since its last application to the classifiers in [A] exceeds a certain threshold θ_{GA} . Then, the system selects two parents from [A]. So far, two selection schemes have been studied: *proportionate selection* [13], in which each classifier in [A] has a probability proportional to its fitness to be chosen; and *tournament selection* [4], in which tournaments are held among a set of randomly selected classifiers and the best classifier of the tournament is chosen as a parent. Then, the parents are crossed and mutated with probabilities χ and μ respectively. Crossover shuffles the condition of the two parents by cutting the chromosomes by two points. Mutation decides whether each variable has to be changed; in this case, it adds a random amount to the lower or to the upper bound of the variable interval. The resulting offspring are introduced into the population, removing potentially poor classifiers if there is not room for them [9].

2. Introducing Evolution Strategies to XCS: Representation and New Operators

Evolution Strategies [12], like GAs, are optimization algorithms that take inspiration from biology to solve complex optimization problems. The main differences between GAs and ESs is that ESs incorporate a vector of strategy parameters that are used by the mutation operator to guide the local search towards the objective. Therefore, mutation is the primary operator of ESs. The vector of strategy parameters is self-adapted during the evolutionary process with the aim of applying more precise mutations to the classifier's conditions. As follows, we explain the new representation, as well as the new genetic operators, proposed to adapt ESs to the interval-based rule representation of XCS.

2.1. Knowledge Representation for Evolution Strategies

Now, the classifier representation is enriched with a vector of strategy parameters $s = (\sigma_1, \sigma_2, \dots, \sigma_\ell)$, which is used to adapt the intervals of the variables of the rules condition (the rules condition is referred to as *object parameters* in ESs terms). Covering initializes each strategy parameter σ_i as $\sigma_i = \text{rand}(0, \mu)$, where μ is initially chosen in the range $[1/N, 1/\ell]$ (N is the population size and ℓ is the number of attributes of the problem). The strategy parameters undergo recombination and mutation during learning; therefore, they self-adapt with the aim of letting mutation perform a more precise local search. All the genetic operators, i.e., selection, mutation, and recombination, are redefined as explained in the following sections.

2.2. Selection

In evolution strategies, the typical selection operator is *truncation selection*, which selects the classifiers of $[A]$ with highest fitness. Since this selection strategy can be quite aggressive, especially in steady state algorithms, we also use proportionate and tournament selection as defined in the previous section.

2.3. Gaussian Mutation

The mutation operator of XCS is redefined as follows. First, we mutate the intervals of each rule variable x_i as $x_i = x_i + z$, where $z = (\sigma_1 N_1(0, 1), \sigma_2 N_2(0, 1), \dots, \sigma_\ell N_\ell(0, 1))$ are independent random samples from Gaussian normal distribution.

The strategy parameters are self-adapted along the XCS run. After mutation, the new vector of strategy parameters s' is updated as $s' = e^{\tau_0 N_0(0, 1)} (\sigma_1 e^{\tau N_1(0, 1)}, \dots, \sigma_\ell e^{\tau N_\ell(0, 1)})$, where τ indicates the precision of self-adaption, τ_0 weights the global effect of mutation, and $N_i(0, 1)$ returns a Gaussian number with $\sigma = 1$. In our experiments, we configured $\tau_0 = 1/\sqrt{2\ell}$ and $\tau = 1/\sqrt{2\sqrt{\ell}}$ as usually done in the ESs literature.

2.4. Recombination

We consider two classes of recombination in ESs: (i) discrete/dominant recombination and (ii) intermediate recombination. Discrete recombination produces a new rule where each variable and strategy parameter is randomly selected from one of the parents. Intermediate recombination calculates the center of mass of the parents; thus, this recombination operator pushes towards the average value per attribute among all classifiers.

3. Experimental Methodology

The main concern of this work is to analyze in detail the effect of the different genetic operators on the genetic search as well as the interaction among themselves. Moreover, the advantages and disadvantages of the new operators will be carefully studied. For this purpose, we start examining XCS's behavior with selection and mutation with the aim of analyzing the search capabilities provided by the combination of the two operators. Then, we add crossover to the study, showing the benefits supplied by this operator. Moreover,

Table 1. Properties of the data sets. The columns describe: the identifier of the data set (Id.) the name of the data set (dataset), the number of instances (#Inst), the total number of features (#Fea), the number of real features (#Re), the number of integer features (#In), the number of nominal features (#No), the number of classes (#Cl), and the proportion of instances with missing values (%MisInst).

| Id. | dataset | #Inst | #Fea | #Re | #In | #No | #Cl | %MisInst |
|-------------|------------------------------|-------|------|-----|-----|-----|-----|----------|
| <i>bal</i> | Balance | 625 | 4 | 4 | 0 | 0 | 3 | 0 |
| <i>bpa</i> | Bupa | 345 | 6 | 6 | 0 | 0 | 2 | 0 |
| <i>gls</i> | Glass | 214 | 9 | 9 | 0 | 0 | 6 | 0 |
| <i>h-s</i> | Heart-s | 270 | 13 | 13 | 0 | 0 | 2 | 0 |
| <i>irs</i> | Iris | 150 | 4 | 4 | 0 | 0 | 3 | 0 |
| <i>pim</i> | Pima | 768 | 8 | 8 | 0 | 0 | 2 | 0 |
| <i>tao</i> | Tao | 1888 | 2 | 2 | 0 | 0 | 2 | 0 |
| <i>thy</i> | Thyroid | 215 | 5 | 5 | 0 | 0 | 3 | 0 |
| <i>veh</i> | Vehicle | 846 | 18 | 18 | 0 | 0 | 4 | 0 |
| <i>wbcd</i> | Wisc. breast-cancer | 699 | 9 | 0 | 9 | 0 | 2 | 2.3 |
| <i>wdbc</i> | Wisc. diagnose breast-cancer | 569 | 30 | 30 | 0 | 0 | 2 | 0 |
| <i>wne</i> | Wine | 178 | 13 | 13 | 0 | 0 | 3 | 0 |

in all the cases, the results of GA-based XCS (referred to as XCS_{GA}) are compared with those obtained with ES-based XCS (referred to as XCS_{ES}), providing some interesting insights on the differences between evolution strategies and genetic algorithms in the context of on-line learning.

For the study, we used a collection of 12 real-life data sets extracted from the UCI repository [1], whose characteristics are summarized in Table 1. The different configurations of XCS_{GA} and XCS_{ES} were ran on these data sets and the quality of the results was compared in terms of the performance (test accuracy) of the final models. To obtain reliable estimates of these metrics we used a ten-fold cross-validation procedure. XCS was configured as follows (see [5] for notation details): $iter. = 100,000$, $N = 6400$, $\theta_{GA} = 50$, $\chi = 0.8$, $\mu = 0.04$, $r_0 = 0.6$, $m_0 = 0.1$.

We statistically analyzed the performance of each learner following the procedure pointed out in [6]. We first applied the multi-comparison Friedman test to contrast the null hypothesis that all the learning algorithms performed the same on average. If the Friedman test rejected the null hypothesis, the post-hoc Bonferroni-Dunn test was used. Moreover, when required, we also applied pairwise comparisons by means of the non-parametric Wilcoxon signed-ranks test.

4. Experimental Results

As follows, we present the results obtained with the combinations of (i) selection and mutation operators and (ii) selection, crossover, and mutation operators.

4.1. Analysis of the Effect of Selection + Mutation

Our first concern was to analyze the behavior of XCS with both GAs and ESs when only the selection operator and the mutation operator were considered. For this purpose, Table 2 supplies the test accuracies and the ranks obtained by XCS_{GA} and XCS_{ES} with

Table 2. Comparison of the test performance and rank obtained with XCS_{GA} and XCS_{ES} with proportionate selection (ps) and tournament selection (ts). Moreover, the results of XCS_{ES} with truncation selection (tr) and weighted XCS_{ES} are also provided. In all the runs, crossover was switched off. The last three rows provide the average performance, the average rank, and the position of each learner in the ranking.

| Dataset | XCS_{GA} -ps | XCS_{ES} -ps | XCS_{GA} -ts | XCS_{ES} -ts | XCS_{ES} -tr | weig. XCS_{ES} |
|-------------|----------------|----------------|----------------|----------------|----------------|------------------|
| <i>bal</i> | 82.35 (2) | 82.08 (3) | 81.55 (5) | 81.71 (4) | 81.12 (6) | 82.93 (1) |
| <i>bpa</i> | 62.51 (5.5) | 64.15 (2) | 62.80 (3) | 65.12 (1) | 62.70 (4) | 62.51 (5.5) |
| <i>gls</i> | 66.51 (6) | 67.13 (4) | 66.98 (5) | 69.94 (1) | 67.29 (3) | 67.91 (2) |
| <i>h-s</i> | 41.23 (4) | 43.46 (1) | 41.60 (3) | 42.72 (2) | 37.78 (6) | 39.63 (5) |
| <i>irs</i> | 94.89 (3) | 93.33 (6) | 95.33 (1) | 94.89 (3) | 94.89 (3) | 94.44 (5) |
| <i>pim</i> | 70.83 (4) | 71.05 (2) | 69.99 (6) | 72.87 (1) | 70.88 (3) | 70.53 (5) |
| <i>tao</i> | 89.32 (6) | 92.90 (3) | 89.79 (5) | 93.80 (1) | 93.01 (2) | 89.90 (4) |
| <i>thy</i> | 94.73 (6) | 95.50 (4) | 95.66 (2.5) | 96.28 (1) | 94.88 (5) | 95.66 (2.5) |
| <i>veh</i> | 65.52 (3) | 66.00 (2) | 64.50 (4) | 67.26 (1) | 63.83 (6) | 64.34 (5) |
| <i>wbcd</i> | 80.88 (6) | 85.84 (1) | 81.26 (5) | 85.65 (2) | 82.50 (4) | 82.93 (3) |
| <i>wdbc</i> | 78.68 (2) | 75.28 (3) | 80.20 (1) | 74.93 (4) | 67.60 (6) | 69.13 (5) |
| <i>wne</i> | 80.71 (5) | 86.70 (1) | 82.21 (2) | 82.02 (3) | 78.09 (6) | 81.65 (4) |
| avg. | 75.68 | 76.95 | 75.99 | 77.26 | 74.55 | 75.13 |
| rnk. | 4.38 | 2.67 | 3.54 | 2.00 | 4.50 | 3.92 |
| pos. | 5 | 2 | 3 | 1 | 6 | 4 |

proportionate and tournament selection (see from the 2nd to the 5th column). Moreover, we also included truncation selection for XCS_{ES} , since it is a selection operator widely used in the ESs field (6th column). The average rank of each learner shows that two schemes based on ESs are the best ranked methods in the comparison. The Friedman test permitted to reject the hypothesis that all the learners were statistically equivalent at $\alpha = 0.01$. The Bonferroni-Dunn test, at $\alpha = 0.05$, indicated that XCS_{ES} with tournament selection significantly outperformed XCS_{GA} with both proportionate and tournament selection. Besides, XCS_{ES} with proportionate selection was significantly better than XCS_{GA} proportionate selection.

Three important observations can be drawn from these results. Firstly, the XCS_{ES} based on truncation selection resulted in the poorest performance of the comparison. We hypothesize that this behavior is because truncation selection is an excessively elitist operator that makes a strong pressure toward the fittest individuals, which goes in detriment of the population diversity. Secondly, the schemes based on tournament selection yielded better results than those schemes based on proportionate selection for XCS_{GA} and XCS_{ES} . These results show the superiority of tournament selection with respect proportionate selection, confirming the theoretical studies presented in [11] and [4]. Therefore, our analysis enables us to extend these conclusions to real-life problems. Thirdly, XCS_{ES} presents brilliant results in the *tao*, *wbcd*, and *wne* data sets, significantly outperforming the results obtained by XCS_{GA} according to a Wilcoxon signed-ranks test at $\alpha = 0.05$. To our knowledge, XCS_{ES} obtained, by far, the best ever reported performance in the *tao* data set, a problem which is especially complicated for XCS since the hyper rectangular representation can barely approximate the decision boundaries of the problem accurately [2].

The overall results indicated that the mutation introduced by XCS_{ES} , i.e., Gaussian mutation, has a greater freedom of action due to a more disruptive behavior introduced

Table 3. Comparison of the test performance and rank obtained with XCS_{GA} and XCS_{ES} with proportionate selection (ps) and tournament selection (ts). Moreover, the results of XCS_{ES} with truncation selection (tr) and XCS_{ES} are also provided. In all runs, we applied selection, crossover, and mutation. The last three rows provide the average performance, the average rank, and the position of each learner in the ranking.

| Dataset | XCS_{GA} -ps | XCS_{ES} -ps | XCS_{GA} -ts | XCS_{ES} -ts | XCS_{ES} -tr |
|-------------|----------------|----------------|----------------|----------------|----------------|
| <i>bal</i> | 83.20 (1) | 82.77 (2.5) | 82.72 (4) | 82.77 (2.5) | 82.13 (5) |
| <i>bpa</i> | 68.21 (1) | 67.05 (2) | 65.22 (4) | 65.70 (3) | 64.06 (5) |
| <i>gls</i> | 72.12 (2) | 71.18 (4) | 73.21 (1) | 71.65 (3) | 69.00 (5) |
| <i>h-s</i> | 46.91 (4) | 51.23 (1) | 47.04 (3) | 49.13 (2) | 44.32 (5) |
| <i>irs</i> | 95.33 (1.5) | 95.33 (1.5) | 94.89 (4.5) | 95.11 (3) | 94.89 (4.5) |
| <i>pim</i> | 72.53 (5) | 74.43 (2) | 73.39 (4) | 73.83 (3) | 74.74 (1) |
| <i>tao</i> | 91.22 (4) | 93.52 (3) | 91.19 (5) | 94.35 (1) | 94.17 (2) |
| <i>thy</i> | 95.81 (2) | 95.50 (5) | 96.43 (1) | 95.66 (3.5) | 95.66 (3.5) |
| <i>veh</i> | 71.79 (2) | 71.75 (3) | 71.20 (4.5) | 72.89 (1) | 71.20 (4.5) |
| <i>wbcd</i> | 94.85 (3) | 95.47 (1.5) | 93.51 (4) | 95.47 (1.5) | 92.61 (5) |
| <i>wdbc</i> | 91.09 (4) | 91.80 (3) | 92.44 (2) | 92.85 (1) | 89.51 (5) |
| <i>wne</i> | 95.50 (4) | 96.25 (1.5) | 95.69 (3) | 96.25 (1.5) | 91.38 (5) |
| avg. | 81.55 | 82.19 | 81.41 | 82.14 | 80.31 |
| rnk. | 2.79 | 2.5 | 3.33 | 2.17 | 4.21 |
| pos. | 4 | 2 | 3 | 1 | 5 |

by the random Gaussian variables. This enables Gaussian mutation to assume tasks of innovation, which has been typically performed by the recombination operator in the GA realm. In order to confirm the hypothesis, we designed a new Gaussian mutation operator, which we addressed as weighed Gaussian mutation. This new operator normalized 95% of the values of Gaussian distribution with the aim of softening the behavior presented by Gaussian mutation. The results obtained with the new operator are shown in the last column of Table 2. The results are similar to those showed by XCS_{GA} , which confirms that non-weighed Gaussian mutation has more freedom of action than random mutation, aspect that promotes the global search capabilities of the system.

4.2. Analysis of the Effect of Selection + Crossover + Mutation

After evaluating the behavior of XCS_{ES} with Gaussian mutation with respect to XCS_{GA} , we now compare the systems with the complete genetic cycle. That is, we ran the same experiments, but adding the crossover operator to each scheme. More specifically, we used two point crossover for XCS_{GA} and a combination of discrete recombination for object parameters and intermediate recombination for strategy parameters for XCS_{ES} . We considered these two crossover methods for each configuration since they were the schemes that maximized the average rank in each case.

Table 3 shows the test accuracy of the different configurations of XCS on the same collection of real-life problems. Several observations can be drawn from the results. Firstly, it is worth noting that the inclusion of crossover improves the test accuracy achieved by XCS in most of the data sets not only for XCS_{GA} , but also XCS_{ES} . Therefore, although theoretical studies that show the benefits of crossover in XCS are lacking, these results support its use to solve complex classification real-life problems. Secondly, as in the previous section, the XCS's schemes based on ESs are the best ranked

in the comparison. The multi-comparison test rejected the null hypothesis that all learners performed the same on average at $\alpha = 0.01$. The post-hoc Bonferroni-Dunn test, at $\alpha = 0.05$, identified that XCS_{ES} with proportionate and tournament selection outperformed XCS_{ES} with truncation selection. No further significant differences were detected. That is to say, differently from the comparison in the previous section, the results obtained by the XCS schemes based on ESs were not significantly better than those based on GAs. This confirms our previous hypothesis that the mutation operator introduced by the ESs permitted a more guided search toward highly fit classifiers in our population, which can be simulated in the GA realm by the use of crossover. Notwithstanding, notice that XCS_{ES} continues being the best ranked learner on average. Finally, let us highlight the brilliant performance presented by XCS_{ES} with any configuration in the *tao* data set, which is even higher than the one obtained in the previous section and than the accuracy reached by XCS_{GA} . In this case, the guidance provided by the evolution strategy seems to be crucial to learn the decision boundaries accurately.

5. Summary, Conclusions, and Further Work

In this paper, we introduced an evolution strategy as a mechanism to discover new promising rules in XCS. We extended the XCS's representation by introducing new parameters required by the evolution strategy and compared XCS with genetic algorithms and XCS with evolution strategies in detail. The effect of the different operators was analyzed carefully. The experimental results evidenced that XCS with evolution strategies presented the best results on average. Moreover, different insights on the role of the different operators were provided.

The overall study presented in this paper served to increase our understanding of how genetic operators work in XCS, especially highlighting the role of crossover and mutation. In addition to all the notes provided while discussing the results, the experimentation highlighted two crucial aspects that should be addressed as further work. Firstly, the results clearly showed that XCS could benefit from new genetic operators. Therefore, more research must be conducted on this regard, designing new operators that consider more information that is available during the genetic evolution. Secondly, results also indicated that different problems benefited from different genetic operators. That is, the performance in problems such as *tao* was impressively increased by the use of an ES. Nonetheless, XCS_{GA} obtained slightly better results than XCS_{ES} in other problems. Therefore, as further work, we will study different strategies to extract characteristics from the training data sets and link these characteristics to the type of operators used during search with the aim of designing hyper-heuristics that enable the system to self-tune its operators depending on the apparent complexity of each particular problem.

Acknowledgements

The authors would like to thank *Enginyeria i Arquitectura La Salle*, Universitat Ramon Llull, as well as the *Ministerio de Educación y Ciencia* for its support under project TIN2005-08386-C05-04 and *Generalitat de Catalunya* for its support under grants 2005FI-00252 and 2005SGR-00302.

References

- [1] A. Asuncion and D. J. Newman. *UCI Machine Learning Repository*: [<http://www.ics.uci.edu/~mllearn/MLRepository.html>]. University of California, 2007.
- [2] E. Bernadó-Mansilla and T.K. Ho. Domain of competence of XCS classifier system in complexity measurement space. *IEEE Transactions on Evolutionary Computation*, 9(1):1–23, 2005.
- [3] M. V. Butz, D. E. Goldberg, and P. L. Lanzi. Gradient descent methods in learning classifier systems: Improving XCS performance in multistep problems. *IEEE Transactions on Evolutionary Computation*, 9(5):452–473, 2005.
- [4] M. V. Butz, K. Sastry, and D. E. Goldberg. Strong, stable, and reliable fitness pressure in XCS due to tournament selection. *Genetic Programming and Evolvable Machines*, 6(1):53–77, 2005.
- [5] M. V. Butz and S. W. Wilson. An algorithmic description of XCS. In P. L. Lanzi, W. Stolzmann, and S. W. Wilson, editors, *Advances in Learning Classifier Systems: Proceedings of the Third International Workshop*, volume 1996 of *Lecture Notes in Artificial Intelligence*, pages 253–272. Springer, 2001.
- [6] J. Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.
- [7] D. E. Goldberg. *Genetic algorithms in search, optimization & machine learning*. Addison Wesley, 1 edition, 1989.
- [8] J. H. Holland. Adaptation. In R. Rosen and F. Snell, editors, *Progress in Theoretical Biology*, volume 4, pages 263–293. New York: Academic Press, 1976.
- [9] T. Kovacs. Deletion schemes for classifier systems. In *GECCO'99: Proceedings of the 1999 Genetic and Evolutionary Computation Conference*, pages 329–336. Morgan Kaufmann, 1999.
- [10] A. Orriols-Puig and E. Bernadó-Mansilla. Evolutionary rule-based systems for imbalanced datasets. *Soft Computing Journal*, 2008.
- [11] A. Orriols-Puig, K. Sastry, P.L. Lanzi, D.E. Goldberg, and E. Bernadó-Mansilla. Modeling selection pressure in XCS for proportionate and tournament selection. In *GECCO'07: Proceedings of the 2007 Genetic and Evolutionary Computation Conference*, volume 2, pages 1846–1853. ACM Press, 2007.
- [12] I. Rechenberg. *Evolution Strategie: Optimierung Technischer Systeme nach Prinzipien der Biologischen Evolution*. Frommann-Holzboog, 1973.
- [13] S. W. Wilson. Classifier fitness based on accuracy. *Evolutionary Computation*, 3(2):149–175, 1995.
- [14] S. W. Wilson. Get real! XCS with continuous-valued inputs. In *Learning Classifier Systems. From Foundations to Applications*, LNAI, pages 209–219, Berlin, 2000. Springer-Verlag.

Intersection and Signed-Intersection Kernels for Intervals

Francisco J. RUIZ ^{a,1}, Cecilio ANGULO ^a and Núria AGELL ^b

^a *Knowledge Engineering Research Group. Universitat Politècnica de Catalunya*

^b *Department of Quantitative Methods Management. ESADE-Universitat Ramon Llull*

Abstract. In this paper two kernels for interval data based on the intersection operation are introduced. On the one hand, it is demonstrated that the intersection length of two intervals is a positive definite (PD) kernel. On the other hand, a signed variant of this kernel, which also permits discriminating between disjoint intervals, is demonstrated to be a conditionally positive definite (CPD) kernel. The potentiality and performance of the two kernels presented when applying them to learning machine techniques based on kernel methods are shown by considering three different examples involving interval data.

Keywords. Qualitative Reasoning, interval analysis, kernel methods

Introduction

In some practical situations, the exact value of a variable is unknown and only an interval of possible values of this variable is available. This happens, for example, if the value is measured by a non-ideal instrument, or if it is the average of several measures. It is also due to the finite nature of computers that cannot cope with the continuous and infinite aspects of real numbers. Many machine learning techniques are specially conceived to deal with exact variables. Specifically the kernel methods, whose best known representative is the Support Vector Machine [2], are initially applied with kernels defined on \mathbb{R}^n , such as polynomial kernels (homogeneous or not), Gaussian kernel, sigmoid kernel, etc. [3]. The kernel formulation is initially used for converting nonlinear problems to linear problems, but due to its nature, these methods are suitable for dealing with any kind of data, even with non-structured data. This is possible by mapping the data into a new space, the feature space. The learning then takes place in this new space. It does not matter the structure of the input space, only the Euclidean structure of the feature space is used.

This characteristic has been used to apply kernel methods with any kind of non-structured data such as biosequences, images, graphs and text documents [4]. The kernel formulation not only allows the use of these sets of data in learning processes, but also permits considering a metric structure and a similarity measure in this set.

In order to use kernel methods with interval data, it is proven that the length of the intersection of two intervals is a positive semi-definite kernel. From the intersection

¹Corresponding Author: EPSEVG-UPC. Avda. Víctor Balaguer, s/n, 08800 Vilanova i la Geltrú; E-mail: francisco.javier.ruiz@upc.edu

length function, another kernel -the Signed Intersection- is also introduced in this paper. The Signed Intersection is not a positive semi-definite kernel but it is a conditionally positive semi-definite kernel suitable for many learning algorithms such as Support Vector Machines. These two kernels presented will allow applying learning machine techniques based on kernel methods when data are described by interval variables.

The rest of the paper is structured as follows. In Section 1 an overview of Kernel Methods is presented. In Section 2 it is demonstrated that the length of the intersection of two intervals is a positive semi-definite kernel. A signed variant of this intersection kernel is also introduced. Section 3 provides samples that illustrate the use of the interval kernels presented in the previous sections using one and two-dimensional data. The last section presents some conclusion and future lines of work.

1. Kernelization

Kernel Methods are a class of algorithms for automatic learning that enable analysis of nonlinear patterns with the efficiency as that of linear methods. The best known Kernel method is the Support Vector Machine (SVM), but also other methods such as Principal Components Analysis (PCA), Fisher Linear Discriminant Analysis (LDA) or the Perceptron algorithm were shown to be capable of operating with kernels. SVM belongs to a family of generalized linear classifiers that maximize the geometric margin between classes. The fact that the formulation of the linear version of SVM and other 'kernelizable' linear methods solely depends on the dot product between input patterns allows replacing this dot product with a suitable function named 'kernel'. Mercer's theorem [6] states that any symmetric and *Positive semi-Definite* (PD) function $K(x,y)$ can be expressed as a dot product in a certain high-dimensional space. The substitution of the dot product by the kernel can be interpreted as mapping the original input patterns into a different higher dimensional space where the linear classifier could separate the classes. The main advantage of kernel functions is that it is not necessary to consider this feature space explicitly and, hence, the dimension of this feature space has no effect on the algorithm.

The use of PD kernels in SVM ensures the Hessian to be positive definite and allows the algorithm to find a unique global minimum value. However, in the SVM algorithm it is possible to use a kernel which does not satisfy Mercer's condition, i.e. non-PD kernels. Even for non-PD kernels, one might still find a positive definite Hessian, in which case the algorithm will converge perfectly.

Conditional Positive semi-Definite (CPD) kernels belong to a larger class of kernels than that of PD kernels. It will emerge that SVM and other learning algorithms work with this class, rather than only with PD kernels. In the same way as PD kernels can be considered as dot products in a certain feature space, CPD can be used as generalized distances in feature spaces [7].

2. Intersection kernel and Signed Intersection kernel

In this section it is proven that the length of the intersection of two intervals is a PD kernel defined on the interval set. Also another function based on the intersection length is proposed to be a kernel, but CPD kernel.

2.1. Intersection Kernel

We first prove that the length of the intersection of two intervals I and J is a PD kernel. In order to prove this, we will consider a feature space formed by the square integrable real functions. The intersection length is equivalent to the usual dot product in this feature space. Once proven to be a kernel, it is not necessary to use the feature space again to calculate $K(I, J)$.

In order to fix the notation that we are going to use in the rest of the paper, we consider two different representation of the bounded and closed interval: the *endpoint* representation, $I = [a, b] = \{x \in \mathbb{R} / a \leq x \leq b\}$ and the *midpoint-radius* representation, $I = B(c, r) = \{x \in \mathbb{R} / |x - c| \leq r\}$.

The set of all the bounded and closed intervals defined on \mathbb{R} is named $I(\mathbb{R})$. By considering closed intervals, it is possible to identify a real number $a \in \mathbb{R}$ with the singleton $[a, a]$, i.e. $\mathbb{R} \subset I(\mathbb{R})$.

Theorem 1 *Let $I_1, I_2 \in \mathbb{R}$ be two bounded and closed intervals. The function*

$$K_{\cap}(I_1, I_2) = \text{length}(I_1 \cap I_2)$$

is a positive definite kernel.

Proof

Let $\phi : I(\mathbb{R}) \longrightarrow L_2(-\infty, +\infty)$ be the mapping onto the Hilbert space of square integrable real functions defined as:

$$\phi(I) = g_{[a,b]} = \begin{cases} 1 & \text{if } a \leq x \leq b \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

with $I = [a, b] \in I(\mathbb{R})$.

The composition of this function with the usual dot product in L_2 ,

$$\begin{aligned} K_{\cap}([a_1, b_1], [a_2, b_2]) &= \int_{-\infty}^{+\infty} g_{[a_1, b_1]}(x) g_{[a_2, b_2]}(x) dx \\ &= \text{length}([a_1, b_1] \cap [a_2, b_2]) \end{aligned} \quad (2)$$

leads to K_{\cap} . ■

2.2. Signed Intersection Kernel

If we consider two intervals $I_1 = B(c_1, r_1)$ and $I_2 = B(c_2, r_2)$ defined by their mid-points and radii, the expression of the Kernel Intersection can be written as:

$$K_{\cap}(I_1, I_2) = \max\{0, \min\{2r_1, 2r_2, r_1 + r_2 - |c_1 - c_2|\}\} \quad (3)$$

In this expression, when the intervals are disjoint, the valor of $r_1 + r_2 - |c_1 - c_2|$ is negative and hence, $K_{\cap}(I_1, I_2) = 0$. This fact makes this kernel inappropriate for discriminating between disjoint intervals. In this section a modification of the Kernel Intersection will be considered. This modification will lead to a simpler and fast-to-compute expression that can also discriminate between disjoint intervals.

Definition Let $I_1 = B(c_1, r_1)$ and $I_2 = B(c_2, r_2)$ be two intervals defined by their midpoints and radii. The *Signed Intersection Kernel* is:

$$K_S(I_1, I_2) = r_1 + r_2 - |c_1 - c_2| \quad (4)$$

The next proposition, will show the relation between this function and the Kernel Intersection.

Proposition 1 Let $I_1, I_2 \in I(\mathbb{R})$ are intervals. It is verified,

1. $K_S(I_1, I_2) \geq K_{\cap}(I_1, I_2)$ for embedded intervals, $I_1 \subset I_2$ or $I_2 \subset I_1$.
2. $K_S(I_1, I_2) \leq K_{\cap}(I_1, I_2)$ for disjoint intervals, $I_1 \cap I_2 = \emptyset$.
3. $K_S(I_1, I_2) = K_{\cap}(I_1, I_2)$ otherwise.

Proof

1. If $I_1 \subset I_2$, then $|c_1 - c_2| \leq r_2 - r_1$, i.e. $r_1 + r_2 - |c_1 - c_2| \geq r_1 + r_2 + r_1 - r_2 = 2r_1$. Hence $K_S(I_1, I_2) \geq K_{\cap}(I_1, I_2)$. A similar proof for $I_2 \subset I_1$ leads to the same result.
2. If $I_1 \cap I_2 = \emptyset$ then $|c_1 - c_2| > r_1 + r_2$. Hence, $K_S(I_1, I_2) \leq 0 = K_{\cap}(I_1, I_2)$. Strict equality appears for the singleton case).
3. For not disjoint, not embedded intervals, length of the (non-empty) intersection is strictly lower than the length of each interval. Hence $K_{\cap}(I_1, I_2) = r_1 + r_2 - |c_1 - c_2| = K_S(I_1, I_2)$. ■

The Signed Intersection is not a PD function, since it does not verify the Cauchy-Schwarz inequality². Nevertheless, the Signed Intersection is conditionally positive semi-definite (CPD). This kind of function belongs to a larger class of kernels that can be considered as a generalized distance in a certain feature space in the same way that PD kernels are dot products in feature spaces [7].

Definition Let $X \neq \emptyset$ a set. A bivariate function $K : X \times X \rightarrow \mathbb{R}$ is a *conditionally positive semi-definite* (CPD) kernel if for all $\{\alpha_1, \dots, \alpha_m\} \in \mathbb{R}$ with $\sum_{i=1}^m \alpha_i = 0$ and $x_i, x_j \in X$, it is satisfied,

$$\sum_{i,j=1}^m \alpha_i \cdot \alpha_j \cdot K(x_i, x_j) \geq 0 \quad (5)$$

In order to demonstrate that the Signed Intersection Kernel is a CPD kernel, we need two previous results.

Lemma 1 $K(\mathbf{x}, \mathbf{y}) = -\|\mathbf{x} - \mathbf{y}\|^2$, with $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ is a CPD kernel.

²If we consider $I_1 = [1, 2]$ and $I_2 = [6, 7]$ then $4^2 = |K_S(I_1, I_2)|^2 > K_S(I_1, I_1) \cdot K_S(I_2, I_2) = 1 \cdot 1$ and, hence, the Cauchy-Schwarz inequality is not satisfied.

Proof

$$\begin{aligned}
\sum_{i,j} \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) &= - \sum_{i,j} \alpha_i \alpha_j \|\mathbf{x}_i - \mathbf{x}_j\|^2 = \\
&= - \sum_i \alpha_i \sum_j \alpha_j \|\mathbf{x}_j\|^2 - \sum_j \alpha_j \sum_i \alpha_i \|\mathbf{x}_i\|^2 + \\
&+ 2 \sum_{i,j} \alpha_i \alpha_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle = \\
&= 0 + 0 + 2 \sum_{i,j} \alpha_i \alpha_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \geq 0, \text{ since } \sum_i \alpha_i = 0. \quad \blacksquare
\end{aligned}$$

Lemma 2 $K(x, y) = -|x - y|$, with $x, y \in \mathbb{R}$ is a CPD kernel.

Proof

It has been demonstrated [1] that for K is a CPD kernel, then $-(-K)^\beta \forall \beta \in (0, 1)$ is also a CPD kernel. Using $\beta = 1/2$ and the latter lemma, $K(x, y) = -|x - y|$ is a CPD kernel. \blacksquare

Theorem 2 The Signed Intersection Kernel K_S is a CPD kernel

Proof

$$\begin{aligned}
\sum_{i,j} \alpha_i \alpha_j K_S(I_i, I_j) &= \sum_{i,j} \alpha_i \alpha_j (r_i + r_j - |c_j - c_i|) = \\
&= \sum_j \alpha_j \sum_i \alpha_i r_i + \sum_i \alpha_i \sum_j \alpha_j r_j + \sum_{i,j} \alpha_i \alpha_j (-|c_j - c_i|) = \\
&= 0 + 0 + \sum_{i,j} \alpha_i \alpha_j (-|c_j - c_i|) \geq 0 \quad \blacksquare
\end{aligned}$$

In [7] and [1] it is argued that CPD kernels are a natural choice whenever they deal with a translation invariant problem, such as the SVM algorithm (maximization of the margin of separation between two classes of data is independent of the origin's position). It will be shown in the next subsection that the Signed Intersection Kernel is useful when the classification problem do no depend on the radius, just on the center.

3. Examples

To illustrate how these interval kernels may be used in learning tasks, three application examples using SVM are considered here.

The first example consists of the classification of subintervals from interval $[0, 1]$ into two classes named, as usual, -1 and 1 . An interval belongs to class 1 if the midpoint belongs to the set $[0.15, 0.40] \cup [0.50, 0.65] \cup [0.80, 0.90]$, whatever their radius. Otherwise belongs to class -1 . In Figure 1a, some intervals of both classes are shown.

In order to better visualize the performance of the two kernels, a diagram called midpoint-radius diagram is used. The midpoint-radius diagram consists of associate to each interval $I = B(c, r)$ the point $(c, r) \in \mathbb{R}^2$. This diagram uses the upper half-plane above the horizontal axis (midpoint axis) to represent all the bounded and closed intervals. The midpoint axis contains all the real numbers.

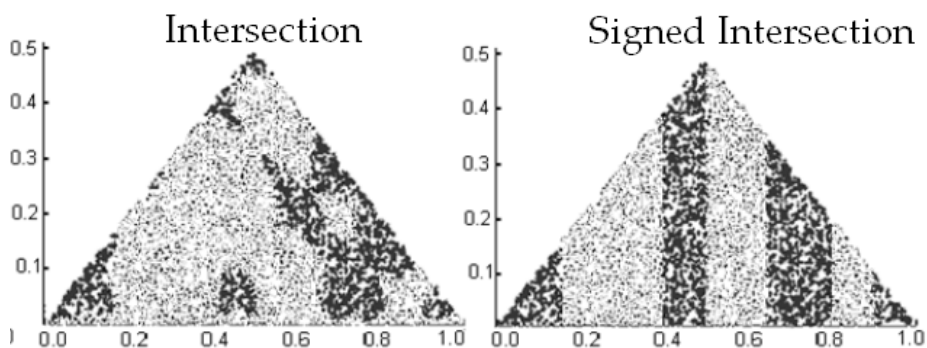
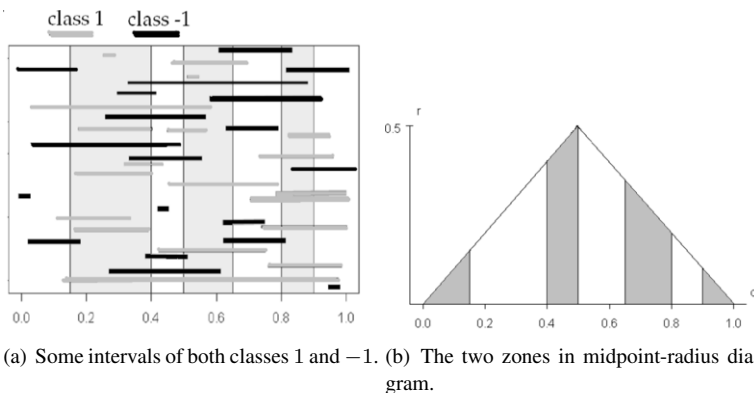


Figure 1. Example 1. Training patterns and results.

Figure 1b represents the two zones in the midpoint-radius diagram that the classifier must find in the first example.

250 patterns were generated for training and 5000 patterns for testing in order to visualize the two classes forecast for the classifier. A SVM was trained using the Intersection Kernel and the Signed Intersection Kernel. By representing the class provided by the classifier with two different colors, the shape of these classes in the midpoint-radius diagram is easily compared (Figure 1c).

By using the Kernel Intersection the test error is 28.20%, this best value is reached using $C = 10$ (Figure 1c left side). In this case, the Kernel Intersection does not reveal a good classification. The best choice in this example is the Signed Intersection Kernel with a test error percentage of 0.80% (with $C = \infty$) (Figure 1c right side). It can be concluded that the Signed Intersection Kernel is more suitable than others when the classification does not depend on the radius but only on the midpoint.

In the second example 250 intervals from $[0, 1]$ are also generated but in this case, we consider as class 1 those intervals most of which, are included in the interval $I_0 = [0.3, 0.5]$, i.e. which belong to the class 1 if $\text{length}(I \cap I_0) \geq \frac{1}{2} \text{length}(I)$.

If c_0 and r_0 are the midpoint and the radius of I_0 , the set of intervals $I = B(c, r)$ belonging to class 1 are those which satisfy $c_0 - r_0 \leq c \leq c_0 + r_0$ and $r \leq 2r_0$.

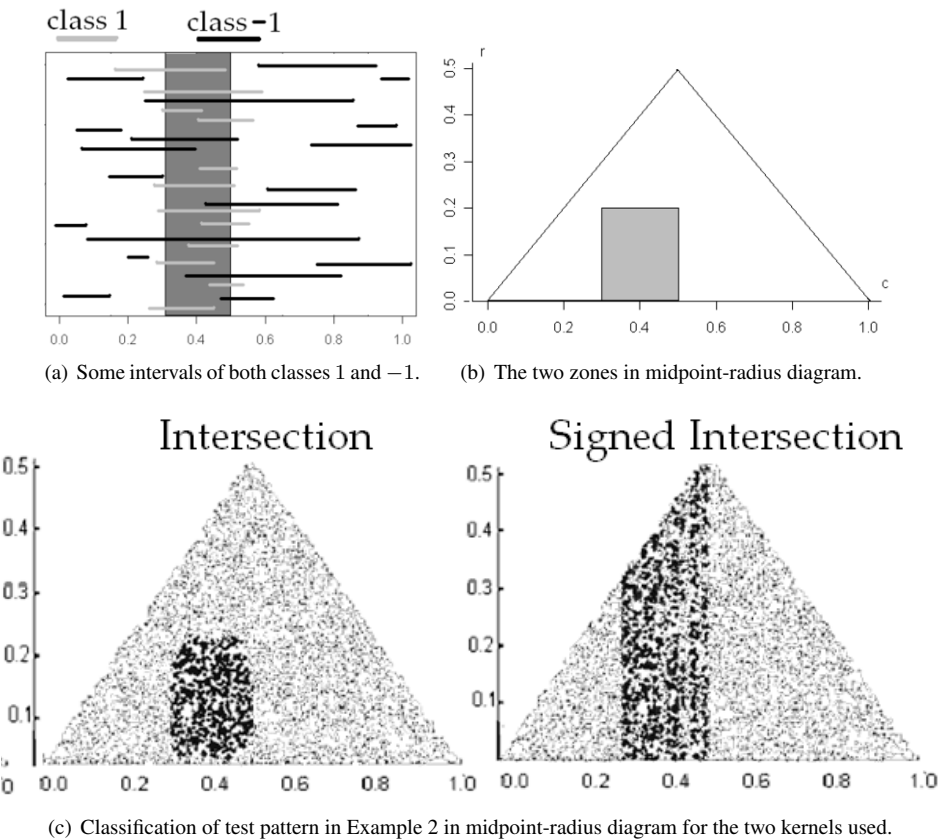
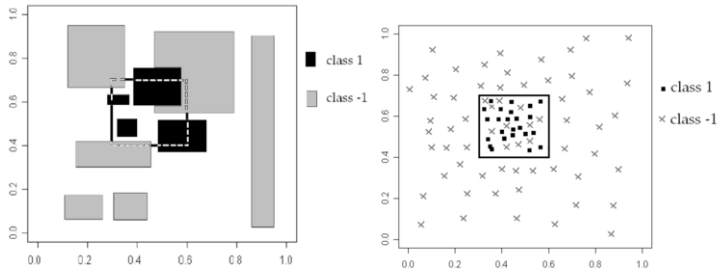


Figure 2. Example 2. Training patterns and results.

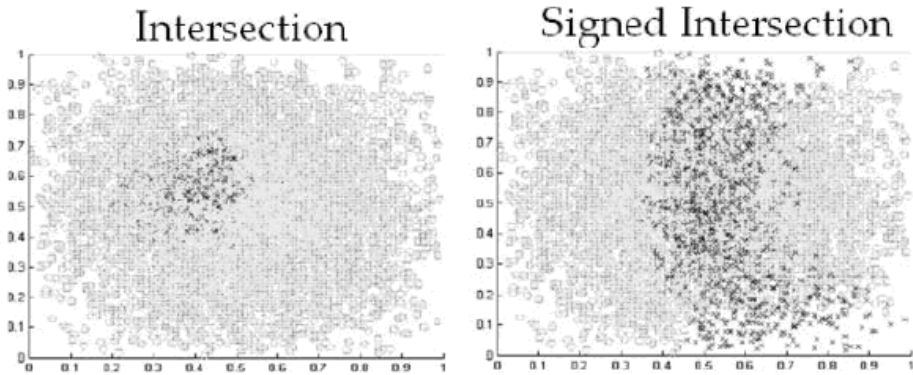
Figure 2a and 2b represents some intervals of both classes and their respective zones in midpoint-radius diagram.

As can be seen in Figure 2c, in this example, the Signed Intersection Kernel (right side) does not give a good classification (error=15.56% with $C = 1000$). However, the Intersection Kernel (left side) gives an acceptable result (error=2.6% with $C = 1000$).

The third example involves two-dimensional interval data. In this case 250 two-dimensional intervals (rectangles) has been generated from the square of opposite vertex $(0, 0)$ and $(1, 1)$. In this case, rectangles from class 1 are those which most of them belong to the rectangle of opposite vertex $(0.3, 0.4)$ and $(0.5, 0.7)$. In Figure 3a some of the training patterns have been represented using two different colors to identify the two classes. In Figure 3b, the midpoint of the training rectangles has been represented in order to visualize the difficulty of the classification. In Figure 3c the midpoint of 5000 test patterns have been represented when the three different kernels have been used. The result using Intersection Kernel was 4.52% ($C = 100$) and the Signed Intersection reached 29.90% ($C = 10$) in error classification. These results reveal that Intersection Kernel performed well. The Signed Intersection Kernel only perform well if the problem does not depend on the precision of the intervals, just on the position.



(a) Some intervals of both classes 1 and -1 . (b) Midpoint of training patterns of both classes.



(c) Classification of test pattern in Example 3 in midpoint-radius diagram for the two kernels used.

Figure 3. Example 3. Training patterns and results.

4. Conclusion

In this paper, we have proven that the length of the intersection of two intervals is a symmetric positive definite function and hence a suitable kernel. On the other hand, when the problem does not depend explicitly on the precision of the interval, another kernel associated to the intersection but non-PD kernel, the Signed Intersection is a good choice. It is important to remark that the concepts presented in this paper can be easily adapted to other kind of data representation. As a future work, it is planned to extend this methodology to more general sets such as fuzzy sets.

Acknowledgements

This research has been partially granted by the projects AURA (TIN2005-08873-C02) and ADA-EXODUS (DPI2006-15630-C02) of the Spanish Ministry of Education and Science.

References

- [1] C. Berg, J.P.R. Christensen and P. Ressel, *Harmonic Analysis on Semigroups*, Springer Verlag, (1984).
- [2] B.E. Boser, I. Guyon and V. Vapnik, A Training Algorithm for Optimal Margin Classifiers, *Computational Learning Theory*, pp 144–152. (1992)
- [3] N. Cristianini and J. Shawe-Taylor, *An introduction to Support Vector Machines and other Kernel-based learning methods*, Cambridge University Press, (2000).
- [4] I. Guyon, *SVM application list*. <http://www.clopinet.com/isabelle/Projects/SVM/applist.html>, Berkeley, (2008).
- [5] Z. Kulpa, Diagrammatic representation for interval arithmetic, *Linear Algebra and its Applications* **324** pp 55–80 (2001).
- [6] J. Mercer, Functions of positive and negative type and their connection with the theory of integral equations. *Philos. Trans. Roy. Soc. London, A* (209) pp 415–446. (1909)
- [7] B. Scholkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond*. MIT Press, (2001).
- [8] V.N.Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, (1995).
- [9] V.N.Vapnik, *Statistical Learning Theory*, John Wiley & sons, (1998).

Multidisciplinary Topics and Applications

This page intentionally left blank

Representation of discrete quasi-copulas through non-square matrices

Isabel AGUILÓ¹, Jaume SUÑER and Joan TORRENS

*Departament de Matemàtiques i Informàtica
Universitat de les Illes Balears
07122 Palma de Mallorca, Spain*

Abstract. In this paper a matrix representation for discrete quasi-copulas defined on a non-square grid $I_n \times I_m$ of $[0, 1]^2$ with $m = kn$, $k \geq 1$ is given. The special case of irreducible discrete quasi-copulas (those that cannot be expressed as non-trivial convex combination of other ones) is also studied. It is proved that quasi-copulas of minimal range admit a matrix representation through a special class of matrices. In the case of copulas, it is proved that irreducible copulas coincide with those of minimal range. Moreover, an algorithm to express any discrete copula as a convex combination of irreducible ones is given.

Keywords. Aggregation function, discrete copula, discrete quasi-copula, matrix representation, irreducibility.

Introduction

Aggregation functions have proved to be essential not only in aggregation data and information fusion, but also in many related applications like group decision making, fuzzy logic and rule based systems, fuzzy preference modelling (see the recent book [3]). A special kind is given by the 1-Lipschitz aggregation functions, useful in some of the mentioned applications. Among them, copulas and quasi-copulas play an important role for describing the structure of 1-Lipschitz aggregation functions with neutral element (see [9]). However, copulas and quasi-copulas are also important mainly for their applications in statistics and probability theory as a consequence of the well known Sklar's theorem (see [18]). They have been extensively studied in relation to this topic (see [2], [18], [17], [6]). On the other hand, the study of copulas and quasi-copulas as aggregation functions has led to interesting results sometimes with new applications from the statistical point of view (see [4], [5], [7], [8], [11], [14]).

An essential concept in the theory of copulas (quasi-copulas) is that of subcopula (defined on a subset of $[0, 1]^2$) and a special case is given by the so-called discrete copulas (quasi-copulas), defined on a finite subset of $[0, 1]^2$, usually $I_n \times I_m$ where, for all $n \geq 1$, $I_n = \{0, \frac{1}{n}, \frac{2}{n}, \dots, \frac{n-1}{n}, 1\}$. The study of discrete copulas and quasi-copulas has been developed in a parallel way usually giving a matrix representation through different kinds of matrices like permutations matrices, bistochastic matrices, alternating sign matrices

¹Corresponding author. E-mail: isabel.aguilo@uib.es

(ASM), and generalized bistochastic matrices (GBM), see successively [13], [10], and [1]. However, all the above mentioned matrix representations are given only for the case $n = m$, and the general case $n \neq m$ has been studied only for copulas in [16].

Discrete copulas defined on $I_n \times I_m$ are interesting because they link uniformly distributed random variables X, Y whose distribution functions have ranges I_n and I_m respectively. But also because they can be applied in aggregation of linguistic variables whose arguments are labels from finite chains such as $\{Very\ Low, Low, Medium, High, Very\ High\}$ (note that in the irreducible case they take values again in I_m). In this line, not only copulas and quasi-copulas have been studied for the discrete case, but also many other aggregation functions like t-norms and t-conorms (see [15]), uninorms and nullnorms (see [12]).

The aim of this paper is to deal with quasi-copulas defined on $I_n \times I_m$ with $n \neq m$ in the special case when $m = kn$ or $n = km$, $k \in \mathbb{Z}^+$ (the general case is quite more complex mainly because different classes of irreducible copulas and quasi-copulas appear, and it will be studied in a future work). We will prove that a matrix representation for discrete quasi-copulas defined on $I_n \times I_m$ works in a similar way as for the case $n = m$. This representation allows to easily obtain all quasi-copulas on $I_n \times I_m$ and also to work with them in a very simple way. The special case of irreducible discrete quasi-copulas (those that cannot be expressed as non-trivial convex combination of other ones) is considered. In this line, quasi-copulas with minimal range are introduced and proved to admit a matrix representation through a special class of matrices. In the case of copulas, it is proved that irreducible copulas coincide with those of minimal range, that any discrete copula on $I_n \times I_m$ is a convex combination of irreducible ones and an algorithm to find this combination is given. The case of quasi-copulas seems to work similarly but it still remains as an open problem.

1. Preliminaries

In this section we give only basic definitions and properties of copulas and subcopulas. A more extensive study can be found in [17] and [6] and, for the discrete case, in [10] and [13].

Definition 1 ([17]) A (two dimensional) copula C is a binary operation on $[0, 1]$, i.e., $C : [0, 1] \times [0, 1] \longrightarrow [0, 1]$ such that

- (C1) $C(x, 0) = C(0, x) = 0 \forall x \in [0, 1]$
- (C2) $C(x, 1) = C(1, x) = x \forall x \in [0, 1]$
- (C3) $C(x, y) + C(x', y') \geq C(x, y') + C(x', y)$
for all $x, x', y, y' \in [0, 1]$ with $x \leq x', y \leq y'$ (2-increasing condition)

Although the notion of quasi-copula was introduced in a different way, the following equivalent definition is proved in [6].

Definition 2 A (two dimensional) quasi-copula Q is a binary operation on $[0, 1]$, i.e., $Q : [0, 1] \times [0, 1] \longrightarrow [0, 1]$ such that

- (Q1) $Q(x, 0) = Q(0, x) = 0 \forall x \in [0, 1]$

- (Q2) $Q(x, 1) = Q(1, x) = x \forall x \in [0, 1]$
 (Q3) Q is non-decreasing in each component
 (Q4) Q satisfies the Lipschitz condition with constant 1:

$$|Q(x', y') - Q(x, y)| \leq |x' - x| + |y' - y| \text{ for all } x, x', y, y' \in [0, 1]$$

Each copula is clearly a quasi-copula but not vice versa. Quasi-copulas which are not copulas are called *proper quasi-copulas*.

Definition 3 ([17]) A function $C : D \times D' \rightarrow [0, 1]$, where D, D' are subsets of $[0, 1]$ containing $\{0, 1\}$, is called a *subcopula* if it satisfies the properties of a copula for all $(x, y) \in D \times D'$.

A special case of subcopulas are the so-called *discrete copulas* defined on a finite subset of $[0, 1]^2$, usually $I_n \times I_m$, where $I_n = \{0, \frac{1}{n}, \frac{2}{n}, \dots, \frac{n-1}{n}, 1\}$. Thus, in our context, we have the following definitions (see, for instance, [10] and [1]).

Definition 4 A discrete copula C on $I_n \times I_m$ is a binary operation $C : I_n \times I_m \rightarrow [0, 1]$ satisfying properties (C1 – C3) for all $x \in I_n$ and $y \in I_m$.

Definition 5 A discrete quasi-copula Q on $I_n \times I_m$ is a binary operation $Q : I_n \times I_m \rightarrow [0, 1]$ satisfying properties (Q1 – Q4) for all $x \in I_n$ and $y \in I_m$.

It is clear that any discrete copula $C : I_n \times I_m \rightarrow [0, 1]$ is a discrete quasi-copula, but not vice versa and, in this sense, we will call *proper discrete quasi-copulas* to discrete quasi-copulas that are not discrete copulas.

In the case $n = m$ the range of a copula C on I_n contains I_n and, when it is exactly I_n , we deal in fact with *internal discrete copulas* $C : I_n \times I_n \rightarrow I_n$, called *irreducible discrete copulas* in [10]. The same notation is used for quasi-copulas in [1] and in both cases it is proved that all discrete copulas (quasi-copulas) are convex combinations of irreducible ones (see again [10] and [1]). Moreover, in this case ($n = m$) discrete copulas allow a representation through bistochastic matrices (permutation matrices for irreducible ones) (see [10] and [13]) whereas discrete quasi-copulas allow an analogous representation through generalized bistochastic matrices and alternating sign matrices for irreducible ones (see [1]).

For the general case of discrete copulas on $I_{n,m}$ it is proved in [16] that a matrix representation is also available as follows. Denote by $\mathbf{M}_{n \times m}$ the set of all $n \times m$ matrices $A_{n,m} = (a_{ij})$ whose entries satisfy:

$$a_{ij} \geq 0, \quad \sum_{i=1}^n a_{ij} = \frac{1}{n}, \quad \sum_{j=1}^m a_{ij} = \frac{1}{m}.$$

Theorem 1 A binary operation $C : I_n \times I_m \rightarrow [0, 1]$ is a discrete copula if and only if there is an $n \times m$ matrix $A_{n,m} = (a_{ij}) \in \mathbf{M}_{n \times m}$ such that

$$C\left(\frac{i}{n}, \frac{j}{m}\right) = \sum_{k=1}^i \sum_{r=1}^j a_{k,r}$$

for all $i = 1, \dots, n$ and for all $j = 1, \dots, m$.

2. Representation of discrete quasi-copulas on $I_n \times I_m$ with $m = kn, k \in \mathbb{Z}^+$

We will suppose from now on, without any loss of generality, that $m \geq n$ (the case $n \geq m$ works in the same way). In fact, we reduce our study to the case $m = kn, k \in \mathbb{Z}^+$.

The general case is more complex and it will be dealt with in a future paper.

Our first goal is to give a matrix representation for discrete quasi-copulas on $I_n \times I_m$ similar to the one stated in [1] for the case $n = m$. For this purpose we introduce the following definition.

Definition 6 $\mathbf{B}_{n \times m}$ is the set of $n \times m$ matrices $A = (a_{ij})$ such that

1. For all $1 \leq i \leq n, 1 \leq j \leq m, a_{ij} \in [-\frac{1}{m}, \frac{1}{m}]$.
2. For all $1 \leq j \leq m, \sum_{i=1}^n a_{ij} = \frac{1}{m},$ for all $1 \leq i \leq n, \sum_{j=1}^m a_{ij} = \frac{1}{n}.$
3. $0 \leq \sum_{i=1}^k a_{ij} \leq \frac{1}{m}, \quad 1 \leq j \leq m, 1 \leq k \leq n,$
 $0 \leq \sum_{j=1}^k a_{ij} \leq \frac{1}{n}, \quad 1 \leq i \leq n, 1 \leq k \leq m.$

Next we give the characterization of quasi-copulas in terms of this class of matrices. We will omit all proofs in the paper due to space limitations, but all of them are similar to those given in [1].

Proposition 1 A binary operation $Q : I_n \times I_m \longrightarrow [0, 1]$ is a discrete quasi-copula if, and only if, there exists an $n \times m$ matrix $A = (a_{ij}) \in \mathbf{B}_{n \times m}$ such that, for all $1 \leq r \leq n, 1 \leq s \leq m,$

$$Q\left(\frac{r}{n}, \frac{s}{m}\right) = \begin{cases} 0 & \text{if } r = 0 \text{ or } s = 0 \\ \sum_{\substack{i \leq r \\ j \leq s}} a_{ij} & \text{otherwise} \end{cases} \quad (1)$$

Given the quasi-copula Q , the matrix A is obtained as

$$a_{ij} = Q\left(\frac{i}{n}, \frac{j}{m}\right) + Q\left(\frac{i-1}{n}, \frac{j-1}{m}\right) - Q\left(\frac{i}{n}, \frac{j-1}{m}\right) - Q\left(\frac{i-1}{n}, \frac{j}{m}\right) \quad (2)$$

Corollary 1 There is a one-to-one correspondence between the set of all discrete quasi-copulas on $I_n \times I_m$ and the set $\mathbf{B}_{n \times m}$. This correspondence assigns to a discrete quasi-copula Q , the matrix $A = (a_{ij})$ given by (2), that will be called from now on the associated matrix of Q .

Note that in this one-to-one correspondence between discrete quasi-copulas on $I_n \times I_m$ and matrices in $\mathbf{B}_{n \times m}$, each negative entry of the associated matrix corresponds to a violation of the 2-increasing condition of copula (exactly as in [1] for the case $n = m$). Thus, it is easy to give examples of proper discrete quasi-copulas Q on $I_n \times I_m$ just by constructing $n \times m$ matrices in $\mathbf{B}_{n \times m}$ with at least a negative entry.

Example 1 A proper quasi-copula Q on $I_3 \times I_6$ and its associated matrix A are the following:

| | | | | | | | |
|---------------|-----|---------------|---------------|---------------|---------------|---------------|---------------|
| Q | 0 | $\frac{1}{6}$ | $\frac{2}{6}$ | $\frac{3}{6}$ | $\frac{4}{6}$ | $\frac{5}{6}$ | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| $\frac{1}{3}$ | 0 | 0 | $\frac{1}{6}$ | $\frac{1}{6}$ | $\frac{2}{6}$ | $\frac{2}{6}$ | $\frac{2}{6}$ |
| $\frac{2}{3}$ | 0 | $\frac{1}{6}$ | $\frac{2}{6}$ | $\frac{2}{6}$ | $\frac{3}{6}$ | $\frac{3}{6}$ | $\frac{4}{6}$ |
| 1 | 0 | $\frac{1}{6}$ | $\frac{2}{6}$ | $\frac{3}{6}$ | $\frac{4}{6}$ | $\frac{5}{6}$ | $\frac{6}{6}$ |

$$A = \begin{pmatrix} 0 & \frac{1}{6} & 0 & \frac{1}{6} & 0 & 0 \\ \frac{1}{6} & 0 & \frac{1}{6} & \frac{1}{6} & 0 & \frac{1}{6} \\ 0 & 0 & 0 & \frac{1}{6} & \frac{1}{6} & 0 \end{pmatrix}$$

As in the particular case $n = m$ we have in general that any convex combination of discrete quasi-copulas is again a quasi-copula, in fact we have the following proposition.

Proposition 2 Let $\{Q_i : i = 1, \dots, k\}$ be a family of discrete quasi-copulas on $I_n \times I_m$. Let α_i be real numbers such that $\alpha_i \geq 0$ and $\sum_{i=1}^k \alpha_i = 1$. Then the binary operation given by $Q(x, y) = \sum_{i=1}^k \alpha_i Q_i(x, y)$, for all $(x, y) \in I_n \times I_m$, is again a quasi-copula. Moreover, if A_i is the associated matrix of Q_i for $i = 1, \dots, k$, then the associated matrix of Q is given by $A = \sum_{i=1}^k \alpha_i A_i$.

From the proposition above one can wonder what discrete quasi-copulas (or their associated matrices) are irreducible, that is, what discrete quasi-copulas cannot be expressed as non-trivial convex combination of other quasi-copulas. Moreover, one can also wonder if any discrete quasi-copula can be expressed as convex combination of irreducible ones and how to find such an expression. We want to deal with these problems in next section.

3. Irreducible discrete quasi-copulas

The questions mentioned previously were solved in the case $n = m$ in [10] and [1] for copulas and quasi-copulas, respectively. In both cases, it is proved that irreducible copulas and quasi-copulas agree with those of minimal range and that the set of all copulas (quasi-copulas) is the smallest convex set (convex closure) containing the irreducible copulas (quasi-copulas).

In our case, when $m = kn, k \in \mathbb{Z}^+$, we have that $I_n \subset I_m$ and since the range of any discrete quasi-copula on $I_n \times I_m$ contains $I_n \cup I_m = I_m$, the minimal range in this case is the proper set I_m . Now, we want to investigate if the results previously mentioned still remain true in our case $m = kn$, for both copulas and quasi-copulas.

Definition 7 A quasi-copula Q (copula C) on $I_n \times I_m$ with range the proper I_m is called a discrete quasi-copula (copula) with minimal range.

Definition 8 An irreducible discrete quasi-copula (copula) on $I_n \times I_m$ is a discrete quasi-copula on $I_n \times I_m$ which can not be expressed as non-trivial convex combination of quasi-copulas (copulas).

Thus, a discrete quasi-copula (copula) on $I_n \times I_m$ with minimal range is an internal binary aggregation function $Q : I_n \times I_m \longrightarrow I_m$ with properties (Q1) – (Q4) (with properties (C1) – (C3)). In this section, we give a characterization of discrete quasi-copulas (copulas) on $I_n \times I_m$ with minimal range in terms of special classes of matrices. Let us deal first with the case of quasi-copulas with minimal range.

Definition 9 $\mathbf{A}_{n \times m}$ is the set of $n \times m$ matrices $A = (a_{ij})$ such that

1. For all $1 \leq i \leq n$, $1 \leq j \leq m$, $a_{ij} \in \{-\frac{1}{m}, 0, \frac{1}{m}\}$.
2. For all $1 \leq j \leq m$, $\sum_{i=1}^n a_{ij} = \frac{1}{m}$; for all $1 \leq i \leq n$, $\sum_{j=1}^m a_{ij} = \frac{1}{n}$.
3. Given any $1 \leq k \leq n$, $0 \leq \sum_{i=1}^k a_{ij} \leq \frac{1}{m}$, for all $1 \leq j \leq m$,
Given any $1 \leq k \leq n$, $0 \leq \sum_{j=1}^k a_{ij} \leq \frac{1}{n}$, for all $1 \leq i \leq n$.

Next we give the characterization of quasi-copulas with minimal range in terms of this class of matrices.

Proposition 3 A binary operation $Q : I_n \times I_m \longrightarrow I_m$ is a discrete quasi-copula (with minimal range) if, and only if, there exists an $n \times m$ matrix $A = (a_{ij}) \in \mathbf{A}_{n \times m}$ such that, for all $1 \leq r \leq n$, $1 \leq s \leq m$,

$$Q\left(\frac{r}{n}, \frac{s}{m}\right) = \begin{cases} 0 & \text{if } r = 0 \text{ or } s = 0 \\ \sum_{\substack{i \leq r \\ j \leq s}} a_{ij} & \text{otherwise} \end{cases} \quad (3)$$

Given the quasi-copula Q , the matrix A is obtained as

$$a_{ij} = Q\left(\frac{i}{n}, \frac{j}{m}\right) + Q\left(\frac{i-1}{n}, \frac{j-1}{m}\right) - Q\left(\frac{i}{n}, \frac{j-1}{m}\right) - Q\left(\frac{i-1}{n}, \frac{j}{m}\right) \quad (4)$$

Corollary 2 There is a one-to-one correspondence between the set of all discrete quasi-copulas on $I_n \times I_m$ with minimal range and the set $\mathbf{A}_{n \times m}$. This correspondence assigns to a discrete quasi-copula Q , the matrix $A = (a_{ij})$ given by (4), that will be called from now on the associated matrix of Q .

Note that the associated matrices of discrete quasi-copulas on I_n^2 with minimal range (that coincide with the irreducibles ones) are the so-called Alternating Sign Matrices (ASM), see [1]. In the general case $n \neq m$, matrices $\mathbf{A}_{n \times m}$ are a generalization of ASM, but in such a way that the alternation of signs can fail.

Example 2 The following two matrices correspond to two discrete quasi-copulas Q_1, Q_2 on $I_3 \times I_6$ with minimal range. The first one with non consecutive negative entries, and the second one with consecutive negative entries:

$$A_1 = \begin{pmatrix} 0 & \frac{1}{6} & 0 & \frac{1}{6} & 0 & 0 \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ 0 & \frac{1}{6} & 0 & \frac{1}{6} & 0 & 0 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & 0 & \frac{1}{6} & \frac{1}{6} & 0 & 0 \\ \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} & \frac{1}{6} \\ 0 & 0 & \frac{1}{6} & \frac{1}{6} & 0 & 0 \end{pmatrix}$$

For the case of copulas with minimal range the situation is the same but with a class of simpler matrices. Specifically:

Definition 10 $\mathbf{A}_{n \times m}^c$ is the set of $n \times m$ matrices $A = (a_{ij})$ such that

1. For all $1 \leq i \leq n$, $1 \leq j \leq m$, $a_{ij} \in \{0, \frac{1}{m}\}$.
2. For all $1 \leq j \leq m$, $\sum_{i=1}^n a_{ij} = \frac{1}{m}$; for all $1 \leq i \leq n$, $\sum_{j=1}^m a_{ij} = \frac{1}{n}$.

It is easy to see that similar results as Proposition 3 and Corollary 2 hold between copulas with minimal range and matrices in $\mathbf{A}_{n \times m}^c$.

Let us now deal with the simpler case of copulas and we begin by proving that any discrete copula C is a convex linear combination of discrete copulas with minimal range. Let us next give the algorithm that, given the $n \times m$ matrix A of a discrete copula C , produces the matrices in $\mathbf{A}_{n \times m}^c$ corresponding to the copulas with minimal range of the convex combination.

Algorithm

Let $A = (a_{ij})$ be the $n \times m$ matrix associated to a copula C on $I_n \times I_m$.

- 1) Let $c_1 = \min\{a_{ij} : a_{ij} \neq 0\}$.
- 2) If $c_1 = \frac{1}{m}$, then $A \in \mathbf{A}_{n \times m}^c$, that is, it corresponds to an irreducible copula and the process has finished.

If not, let us construct a matrix $A_1 \in \mathbf{A}_{n \times m}^c$ with only the following restrictions: We put a $\frac{1}{m}$ in position (i, j) where c_1 is reached and 0's in all positions where A has 0's.

If the minimum c_1 is reached in more than one position, then we can put a $\frac{1}{m}$ in any one of these positions.

- 3) Let $A_1^* = \frac{1}{1 - mc_1}(A - mc_1 A_1)$. Note that equivalently, we will have

$$A = mc_1 A_1 + (1 - mc_1) A_1^*.$$

A_1^* corresponds to a copula on $I_n \times I_m$. If it is a matrix corresponding to a copula with minimal range, the algorithm is finished. If not, we apply steps 1, 2 and 3 to the matrix A_1^* .

This process ends up with the list of matrices in $\mathbf{A}_{n \times m}^c$ and the coefficients of the convex linear combination are easily obtained from the algorithm.

The crucial point in the algorithm is the fact, in step 3, that the matrix A_1^* correspond to a copula on $I_n \times I_m$. Let us prove this fact with the following lemma.

Lemma 1 For any matrix $A = (a_{ij}) \in \mathbf{M}_{n \times m}$ the corresponding matrix A_1^* given in the previous algorithm corresponds to a copula on $I_n \times I_m$, that is, A_1^* also lies in $\mathbf{M}_{n \times m}$.

Proof: Since in both matrices A and A_1 the sum of each row is equal to $\frac{k}{m} = \frac{1}{n}$ and the sum of each column is $\frac{1}{m}$, the same occurs for the matrix A_1^* . Thus, it only remains to prove that all entries in A_1^* are in $[0, \frac{1}{m}]$. We do it in some steps:

- If the entry a_{ij} of A is 0, the corresponding entry of A_1 is also 0 and so is the entry of A_1^* .
- If the entry $a_{ij} \neq 0$, let us denote by a_{ij}^* the corresponding entry of A_1^* . Then, we have on one hand that

$$a_{ij}^* \geq \frac{1}{1 - mc_1} a_{ij} - \frac{mc_1}{1 - mc_1} \frac{1}{m} = \frac{1}{1 - mc_1} (a_{ij} - c_1) \geq 0$$

because of the definition of c_1 . On the other hand, it can not be $\frac{1}{1 - mc_1} a_{ij} > \frac{1}{m}$ because otherwise

$$\frac{1}{m} - a_{ij} < \frac{1}{m} - \frac{1 - mc_1}{m} = c_1$$

proving that there should be an entry in the column j smaller than c_1 which is a contradiction. Thus, we obtain

$$a_{ij}^* \leq \frac{1}{1 - mc_1} a_{ij} \leq \frac{1}{m}.$$

q.e.d.

As a consequence of the previous algorithm we have that irreducible copulas on $I_n \times I_m$ coincide with those of minimal range, and that any discrete copula is a convex combination of irreducible discrete copulas. That is,

Proposition 4 a) *The class of all irreducible discrete copulas coincides with the class of discrete copulas with minimal range.²*

b) *The class of all discrete copulas on $I_n \times I_m$ is the smallest convex set containing the class of all irreducible discrete copulas.*

Example 3 *Let us consider the copula $C : I_3 \times I_9 \rightarrow [0, 1]$ given by the following 3×9 matrix:*

$$A = \begin{pmatrix} \frac{1}{27} & 0 & \frac{1}{27} & \frac{1}{27} & 0 & \frac{3}{27} & 0 & 0 & \frac{3}{27} \\ \frac{2}{27} & \frac{3}{27} & 0 & \frac{1}{27} & \frac{2}{27} & 0 & \frac{1}{27} & 0 & 0 \\ 0 & 0 & \frac{2}{27} & \frac{1}{27} & \frac{1}{27} & 0 & \frac{2}{27} & \frac{3}{27} & 0 \end{pmatrix}$$

The first step of the algorithm gives $c_1 = \frac{1}{27}$, reached for instance in position a_{11} . Thus, we can choose the following matrix $A_1 \in \mathbf{A}_{3 \times 9}^c$:

$$A_1 = \begin{pmatrix} \frac{1}{9} & 0 & 0 & 0 & 0 & \frac{1}{9} & 0 & 0 & \frac{1}{9} \\ 0 & \frac{1}{9} & 0 & \frac{1}{9} & \frac{1}{9} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{9} & 0 & 0 & 0 & \frac{1}{9} & \frac{1}{9} & 0 \end{pmatrix}$$

Then, the matrix $A_1^ = \frac{1}{1 - 9c_1}(A - 9c_1 A_1) = \frac{3}{2}A - \frac{1}{2}A_1$ is*

²This result is no longer true in the general case when $m = kn + r$ with $0 < r < n$ and this fact leads to a more complex situation.

$$A_1^* = \begin{pmatrix} 0 & 0 & \frac{1}{18} & \frac{1}{18} & 0 & \frac{2}{18} & 0 & 0 & \frac{2}{18} \\ \frac{2}{18} & \frac{2}{18} & 0 & 0 & \frac{1}{18} & 0 & \frac{1}{18} & 0 & 0 \\ 0 & 0 & \frac{1}{18} & \frac{1}{18} & \frac{1}{18} & 0 & \frac{1}{18} & \frac{2}{18} & 0 \end{pmatrix}$$

A_1^* is the matrix of a copula on $I_3 \times I_9$, but $A_1^* \notin \mathbf{A}_{\mathbf{3} \times \mathbf{9}}^{\mathbf{c}}$. Thus we apply steps 1, 2 and 3 to this matrix. Now we have $c_2 = \frac{1}{18}$, reached in a_{13} , and we choose an appropriate $A_2 \in \mathbf{A}_{\mathbf{3} \times \mathbf{9}}^{\mathbf{c}}$, obtaining the corresponding matrix $A_2^* = 2A_1^* - A_2$:

$$A_2 = \begin{pmatrix} 0 & 0 & \frac{1}{9} & 0 & 0 & \frac{1}{9} & 0 & 0 & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & 0 & 0 & \frac{1}{9} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{9} & 0 & 0 & \frac{1}{9} & 0 & 0 \end{pmatrix}, \quad A_2^* = \begin{pmatrix} 0 & 0 & 0 & \frac{1}{9} & 0 & \frac{1}{9} & 0 & 0 & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & 0 & 0 & 0 & 0 & \frac{1}{9} & 0 & 0 \\ 0 & 0 & \frac{1}{9} & 0 & \frac{1}{9} & 0 & 0 & \frac{1}{9} & 0 \end{pmatrix}$$

Now, A_2^* results to be a matrix in $\mathbf{A}_{\mathbf{3} \times \mathbf{9}}^{\mathbf{c}}$ and thus the algorithm has finished. Finally, we obtain the convex combination:

$$A = \frac{1}{3}A_1 + \frac{2}{3}A_1^* = \frac{1}{3}A_1 + \frac{2}{3}\left(\frac{1}{2}A_2 + \frac{1}{2}A_2^*\right) = \frac{1}{3}A_1 + \frac{1}{3}A_2 + \frac{1}{3}A_2^*.$$

Consequently, the discrete copula C is given by the convex linear combination:

$$C = \frac{1}{3}C_1 + \frac{1}{3}C_2 + \frac{1}{3}C_2^*$$

where C_1, C_2 and C_2^* are the irreducible discrete copulas with associated matrices A_1, A_2 and A_2^* , respectively.

For the case of quasi-copulas all the same arguments will work if we can find a similar algorithm as in the case of copulas. The natural generalization of the mentioned algorithm would be to consider $c_1 = \min\{|a_{ij}| : a_{ij} \neq 0\}$ and then construct the matrix $A_1 \in \mathbf{A}_{n \times m}$ with only the following restrictions: We put a $\frac{1}{m}$ (if $a_{ij} > 0$) or a $\frac{-1}{m}$ (if $a_{ij} < 0$) in position (i, j) and 0's in all positions where A has 0's. If the minimum c_1 is reached in more than one position, then we can put a $\frac{1}{m}$ or a $\frac{-1}{m}$ (depending on whether $a_{ij} > 0$ or < 0) in any one of these positions. Finally, define A_1^* as in the case of copulas. Unfortunately, this algorithm does not work in general because A_1^* needs not be the matrix associated to a quasi-copula on $I_n \times I_m$. Specifically, there can be entries a_{ij} in A such that $\frac{1}{1-mc_1} a_{ij} > \frac{1}{m}$. In such cases we would need to put a $\frac{1}{m}$ in the corresponding entry of A_1 . However this is not always possible since there can be more than k positions in matrix A with this situation, and only k values $\frac{1}{m}$ can be put in matrix A_1 since the sum of each row must be $\frac{k}{m} = \frac{1}{n}$ (see the following example).

Example 4 Let us consider the matrix

$$A = \begin{pmatrix} 0 & \frac{4}{60} & \frac{4}{60} & 0 & \frac{4}{60} & \frac{4}{60} & 0 & \frac{4}{60} & 0 & 0 & 0 & 0 \\ \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & \frac{5}{60} & 0 \\ 0 & \frac{3}{60} & \frac{3}{60} & 0 & \frac{3}{60} & \frac{3}{60} & 0 & \frac{3}{60} & 0 & 0 & 0 & \frac{5}{60} \end{pmatrix}$$

This is a matrix in $\mathbf{B}_{\mathbf{3} \times \mathbf{12}}$. Note that in this case $k = 4$ but all the 5 entries in the first row of A are such that $\frac{1}{1-12c_1} a_{ij} > \frac{1}{12}$. That is, there will be one entry in A_1^* greater than $\frac{1}{12}$ and consequently A_1^* does not correspond to a quasi-copula on $I_3 \times I_{12}$.

Thus, a natural question arises: Is it possible to modify the previous algorithm in a way that it runs for quasi-copulas? We claim that the answer is positive. On the other hand, note that a negative answer would mean that there are irreducible quasi-copulas on $I_n \times I_m$, where $m = kn$, other than those of minimal range.

Acknowledgments

This paper has been partially supported by the Spanish grants MTM2006-05540 and MTM2006-08322, and the Government of the Balearic Islands grant PCTIB-2005GC1-07.

References

- [1] I. Aguiló, J. Suñer and J. Torrens: Matrix representation of discrete quasi-copulas, *Fuzzy Sets and Systems* **159** (2008), 1658–1672.
- [2] C. Alsina, R.B. Nelsen and B. Schweizer: On the characterization of a class of binary operations on distribution functions, *Statist. Probab. Lett.* **17** (1993), 85–89.
- [3] G. Beliakov, A. Pradera and T. Calvo: Aggregation Functions: A Guide for Practitioners, *Studies in Fuzziness and Soft Computing* **221**, Springer-Verlag, Heidelberg, 2007.
- [4] F. Durante: Solution of an open problem for associative copulas, *Fuzzy Sets and Systems* **152**(2) (2005), 411–415.
- [5] F. Durante, R. Mesiar and C. Sempì: On a family of copulas constructed from the diagonal section, *Soft Computing* **10**(6) (2006), 490–494.
- [6] C. Genest, J.J. Quesada Molina, J.A. Rodríguez Lallena and C. Sempì: A characterization of quasi-copulas, *Journal of Multivariate Analysis* **69** (1999), 193–205.
- [7] E.P. Klement and A. Kolesárová: Extension to copulas and quasi-copulas as special 1-Lipschitz aggregation operators, *Kybernetika* **41**(3) (2005), 329–348.
- [8] E.P. Klement and A. Kolesárová: Intervals of 1-Lipschitz aggregation operators, quasi-copulas, and copulas with given affine section, *Monatshefte für Mathematik* **152** (2007), 151–167.
- [9] A. Kolesárová: 1-Lipschitz aggregation operators and quasi-copulas, *Kybernetika* **39** (2003), 615–629.
- [10] A. Kolesárová, R. Mesiar, J. Mordelová and C. Sempì: Discrete copulas, *IEEE Transaction on Fuzzy System* **14**(5) (2006), 698–705.
- [11] A. Kolesárová and J. Mordelová: Quasi-copulas and copulas on a discrete scale, *Soft Computing* **10** (2006), 495–501.
- [12] M. Mas, G. Mayor and J. Torrens: t -operators and uninorms on a finite totally ordered set, *Int. J. of Intelligent Systems* **14** (1999), 909–922.
- [13] G. Mayor, J. Suñer and J. Torrens: Copula-like operations on finite settings, *IEEE Transaction on Fuzzy Systems* **13**(4) (2005), 468–477.
- [14] G. Mayor, J. Suñer and J. Torrens: Sklar's Theorem in Finite Settings, *IEEE Transactions on Fuzzy Systems* **15**(3) (2007), 410–416.
- [15] G. Mayor and J. Torrens: Triangular norms in discrete settings, in: E.P. Klement and R. Mesiar (Eds.), *Logical, Algebraic, Analytic, and Probabilistic Aspects of Triangular Norms*. Elsevier, Amsterdam, 2005, 193–236.
- [16] J. Mordelová and A. Kolesárová: Some results on discrete copulas, *Proceedings of EUSFLAT-2007* (2007), 145–150.
- [17] R.B. Nelsen: An introduction to copulas, Springer-Verlag, *Lecture Notes in Statistics* **139**, New York, 1999.
- [18] A. Sklar: Fonctions de répartition à n dimensions et leur marges, *Publ. Inst. Statist. Univ. Paris* **8** (1959), 229–231.

A system to extract social networks based on the processing of information obtained from Internet

Xavi CANALETA^a, Pablo ROS^a, Alex VALLEJO^b, David VERNET^c, and Agustín ZABALLOS^b

^a*Grup de Recerca en Sistemes Distribuïts*

^b*Secció de Telemàtica*

^c*Grup de Recerca en Sistemes Intel·ligents*

Enginyeria i Arquitectura La Salle – Universitat Ramon Llull

{xavic,pros,avallejo,dave,zaballos}@salle.url.edu

Abstract. The amount of information available on Internet increases by the day. This constant growth makes it difficult to distinguish between information which is relevant and which is not, particularly in key areas which are of strategic interest to business and universities or research centres. This article presents an automatic system to extract social networks: a software designed to generate social networks by exploiting information which is already available on Internet through the use of common search engines such as Google or Yahoo. Furthermore, the metrics used to calculate the affinities between different actors in order to confirm the real ties which connect them and the methods to extract the semantic descriptions of each actor are presented. This represents the innovative part of this research.

Keywords. Social networks, web mining, web crawling, affinity.

Introduction

A *social network* is defined as a specific group of actors who are linked to each other through a series of social ties. The strength of these ties may vary depending on the actors in question.

A social network can be configured by obtaining information from each of the individual actors who might be in some way connected. Surveys carried out on different individuals, a behavioural analysis of the community and a data collection of each candidate provide a topology of the social network and show the strength of relation between the actors.

A clear example of the configuration of social networks is the creation of virtual communities by means of Internet portals. The appearance of Web 2.0 has placed a much greater emphasis on these user groups. The origin of this type of networks dates back to 1995 when Randy Conrads created Classmates. The year 2002 saw the beginning of web sites which promoted the creation of communities of friends online. And, as from 2003, the emergence of Friendster, MySpace or LinkedIn has made these types of sites popular. Currently, one of the most popular social networks in Spain is

Facebook. But the extracting system of these classic networks is not automatic. That is, the information required from each actor is provided by the individual himself.

The purpose of our line of work is that the extracting system of the social network is automatic. The information required from each actor is provided by the system which is fed in an automated form by Internet to compile previously stored data. The main difficulty lies in ensuring that the information extracted from each actor is of good enough quality. To achieve this, we have to address the problems related to the immense quantity of information which is difficult to classify [1], to the difficulty of identifying the actors [2], and the use of appropriate formulae to connect the actors of the social network to establish corresponding degree of affinity.

The remainder of this paper is structured or organized as follows. Section 1 provides the related work in this area. Section 2 describes our system called SNex and the extraction process. Section 3 is about the visualization of the social network. Section 4 presents the results and conclusions and, finally, Section 5 describes the future work.

1. Antecedents

The first system dedicated to the automatic extraction of information was the Referral Web by Kautz, Selman, and Shah [3] in 1997. This system was designed to extract social networks through the search engine Altavista. The social network generated was ego-centric, i.e. it was centred on a determined individual. The basic idea of Referral Web is based on the theory of the six degrees of separation and the list of potential individuals related to the original actor was refined using the Jaccard coefficient [4] to obtain the affinity calculations. The main problem was the ambiguity of the names and an attempt was made to resolve this by adding extra fields for the user to clarify names.

A more recent system (presented in the sector) is that of Flink de Mika [5], which provides an in-depth study of the conclusions of Referral Web and improves with additional functions. The architecture of Flink is divided into three layers: data acquisition, storage and visualisation. The system was capable of visualising social networks of semantic network communities. The data obtained was from the web, from the FOAF profiles, from emails and from published articles. The web-mining process employed by Flink is similar to that of the Referral Web and it also uses the Jaccard coefficient [4].

Further work worth mentioning is Polyphonet by Hamasaki and Matsuo [6], also based on [3]. The system works from a list of names by extracting the social network related to the data which the search engine Google provides. Unlike the previous systems mentioned, Polyphonet uses the Overlap coefficient to calculate the affinity between actors. Polyphonet also incorporates methods to detect the type of relation between individuals, which might include co-authoring, participation in joint projects, etc. It also tries to improve scalability by reducing the number of hits on the search engine in order to reduce the computational costs of the extracting system.

The work carried out by De la Rosa and Martínez Gasca [2], which analyses the results of Polyphonet and Flink, should also be highlighted. A methodology to extract social networks is presented where electronic mails of the hits on the search engine are used to construct the extraction process.

SNex, the system presented, is a continuation of their most recent lines of research. On the one hand, the extracting system is fed by the information obtained from the hits

on Google and Yahoo, based on the Polyphonet models. On the other hand, the base for information searches is based on the emails of the actors, as proposed by De la Rosa [2]. We must also consider the scalability parameters to optimise temporary costs. In the system, any of the coefficients described in the previously mentioned works can be employed to calculate affinities. Furthermore, the system incorporates new metrics which can be used to assess the affinity between actors by referring to the semantic descriptors extracted from the information obtained from Internet. This innovating aspect differentiates the system from the previously mentioned works.

2. Description of the System

The extraction process of a social network has three clearly differentiated phases:

1. Web crawling: tracks all the information available an actor. The crawler takes advantage of the capacity of existing search engines to carry out this process.
2. Web mining: extraction process, which uses the information gathered from Internet and the semantic content of each actor as its starting point. New actors linked to the original actor who will in turn expand the social network are obtained from this information
3. Visualisation: navigation tools which enable us to browse on the social web and to consult all the stored information.

The typology of the social networks (researchers who usually belong to a university) to be extracted has been restricted in this application, as have the search techniques and the type of content to be used in the web mining process.

2.1. Web Crawling

The first step when making a query on the search engines is to determine the access technique in order to carry out a consultation. Two search engines were used in this work: Google and Yahoo. The access to the search engine Yahoo was made through its API [7]. In the case of Google queries were made directly through the search engine and a parser was created with the results.

Once access to the search engines had been prepared, the next step was to make the queries. The ambiguity of the results of the searcher when it operates with names and surnames was affirmed. The previously mentioned studies, [5, 6], attempt to resolve this problem with different techniques. In the case of Polyphonet, the name of the actor and keywords to differentiate one actor from another are added. However, given the research environment, this technique would not be considered a valid solution since, on one hand, it involves finding out information about the actor (something which is not always possible when we are dealing with an expansion of the social network) and, on the other, in tests carried out, the keywords used for the main actor did not always succeed in eliminating ambiguity.

One approach to resolve the problem is that described by De la Rosa [2] which uses the actor of the email as an identifier. The results obtained by the searcher in the tests carried out match the requested actor 100%.

Finally, an attempt was made to restrict the information from Internet to that which was strictly relevant to the social network and we aimed to reduce the results which do

not provide such information. Information of interest to researchers in the academia is found in articles and published work (academic papers). Information on the areas in which they develop their work and further information about the subjects themselves can be found in such articles.

This type of document can often be found as published works in Internet and is usually in PDF format due to its portability. At the moment this affirmation may well be true. However, it is nevertheless limited to the majority of subjects concerning Computer Science and not necessarily the case for the rest of the university community. It is expected that advances in Information and Communication Technologies (ICT) will make articles such as the aforementioned on any subject matter available in the near future. Therefore, taking advantage of the parameters of the search engines, the only information from PDF documents will be retrieved.

2.2. Web Mining

Data extraction of the actors is the most crucial point of the process. Posterior analysis to determine the social network and the ties between the different actors will depend on this stage. The objective is to obtain information on an actor (relation of the actor and his semantic description through keywords) from the results obtained by the searcher, from the list of URLs associated to the actor.

SNex

Llista d'actors a analitzar:

Email: @

Email: @

Email: @

Email: @

Email: @

Paràmetres de configuració

Profunditat Xarxa Social

Nivells:

Motor de cerca

Google: ☐

Yahoo: ☐

Càlcul Afinitat

Coefficient Jaccard: ☐

Coefficient Overlap: ☐

Coefficient Cosinus: ☐

Filtres e-mails

Màx emails per document:

Número emails per densitat:

Caracters proximitat:

Dominis

Agrupació soundex: ☐

Caché

Refer caché disc: ☐

Refer Actors: ☐

Refer URLs Spider: ☐

Refer Paraules Clau: ☐

Refer Tot: ☐

Crear Xarxa Social!

Figure 1. SNex 1.0 configuration environment

The *mining* process has different stages: social network expansion, extraction of the semantic descriptions of the actors and, finally, the affinity calculations between the different actors who form part of the social network. The decisions taken at each of these stages are described below.

2.2.1. Social Network Expansion

The primary objective is to be able to expand the social network of the actor who is being analysed. It is important to identify all the actors related to the individual to be tested at this point. To achieve this list of PDF documents related to the individual being tested which the searcher has sent back is referred to.

The first step is to transform the PDF file to a plain text document. The extraction of candidates related to the actor being tested then proceeds. To undertake this task, the decisions taken to identify the valid candidates on completion of the tests will be described:

- Every email which appears in the content of the document is converted into a potential tie with the actor being tested
- Every email which appears in the same line as the actor being tested will be considered as being of greater strength, especially if they are co-authors (affinity value equals 1)
- The emails found near the email of the actor being questioned will be considered as a direct relation (affinity=1). An e-mail at a distance of n characters (configurable in SNex) from the actor being tested is considered as near.
- All the emails that belong to a document with a density of electronic mail superior to d (configurable in SNex) will be penalised and will be of a lesser importance when calculating the affinity with the actor being tested. In this case, the affinity is multiplied by 0.2.
- Finally, all the e-mails which appear in document containing more than m emails (configurable in SNex) will be rejected. This measure ensures that documents which contain sending lists which do not provide any relevant information are deleted.

The application of these rules provides us with a list of candidates to expand the social network towards the next level. The acceptance of these candidates as members of the network will also depend on the affinity calculations made.

2.2.2. Extraction of Semantic Descriptions

The method used to extract the semantic content related to the actor being tested employs well-known text mining procedures, based on the NLP-IR system (Natural Language Process - Information Recovery).

The first step is to carry out a filter of the document in which all words, symbols and characters not providing useful information are deleted.

The second stage is to delete all the *stopwords* (words which do not provide us with any information, such as prepositions, articles, etc.) from the filtered text. To achieve this, the document language must be known. A procedure based on the deletion of *stopwords* has been developed. Previously created lists of *stopwords* generated from the grammatical and linguistic study of [8] are used to count the number of *stopwords* in the document. The highest number of *stopwords* found determines the language of the document.

Finally, in the third phase, a *stemmer* (or grouper) is in charge of grouping together words based on their root. Once this process has been completed, the lemmatisation, i.e. the representation through a single term of all the inflected forms of a word [9] is carried out. One of the first *stemming* algorithms was developed in 1980 by Martin Porter [10], hence the name, "the Porter stemming algorithm." In our system, we used two stemming algorithms for Spanish and the Porter Stemmer for English [11] and the stemming algorithm for Galician created by Penbad [12].

2.2.3. Affinity Calculations

Once a selection of candidates related to the actor being tested has been confirmed and the semantic descriptions of the content of the associated documents extracted, it is now time to decide whether the ties between the actors do really exist and if these ties are strong enough to take into consideration.

The aforementioned works [5, 6], which analyse the number of documents which link the different actors, have been used as a base to calculate the ratio capable of quantifying the degree of affinity between actors. The possibility of calculating affinity with the Jaccard coefficient [4], employed by Flink and Referral Web, the Overlap coefficient, employed by Polphonet, as well as the Cosinus Coefficient have all been contemplated in the system. The idea is to configure a weighted combination of the different coefficients in order to obtain a more reliable affinity calculation, although at present, the weights of the coefficients is still under testing.

The Jaccard coefficient gives the relation between the documents in which both actors appear in reference to the number of documents written by one actor or the other (either of the actors). The Overlap coefficient relates the number of documents jointly written by X and Y, between the minimum of the results produced by actors X and Y.

The Overlap coefficient compensates for the limits of the Jaccard coefficient which occurs when author X has many documents while Y has few but all of Y's documents are linked to X. The Jaccard coefficient provides a value which is of little relevance. The Overlap coefficient rectifies this deficiency.

The Cosinus coefficient resolves the deficiencies of the Overlap coefficient which is produced when, for example, actor X has 100 documents and actor Y has only one document but this document is shared with actor/author X. If we apply the Overlap coefficient the affinity is maximum whereas with the Cosinus coefficient it is 0.1.

$$Jaccard = \frac{n_{X \cap Y}}{n_{X \cup Y}} \quad Overlap = \frac{n_{X \cap Y}}{\min(n_X, n_Y)} \quad Cos = \frac{n_{X \cap Y}}{\sqrt{n_X \cdot n_Y}}$$

From these affinity calculations and taking into account the criteria described in section 2.2.1, the candidates to form part of the social network are filtered. To complete this filter, an affinity value of 0.2 has been determined as a threshold.

Another type of affinity between actors is that which can be obtained through the semantic descriptions extracted from each of the individuals who make up the social network. This method enables us to link actors who may not share a high degree of affinity but who do work in similar areas and consequently put them in touch with each other.

To carry out this calculation, a list of keywords has been selected for each actor. The keyword list is made up of keywords taken from their documents, selecting the words with the highest occurrence in the text. In the SNex Version 1.0, the 10 most frequent words are selected for each actor. This information enables us to develop the search of a formula or metric which can be used to calculate the semantic affinity. The keyword groups of each actor are defined to carry out this task.

$$L_X = \{\text{stem}_0, \text{stem}_1, \dots, \text{stem}_n\} \quad L_Y = \{\text{stem}_0, \text{stem}_1, \dots, \text{stem}_n\}$$

An initial formula has been developed to calculate semantic affinity. The theory behind it is based on the calculation of the number of keywords which coincide with both actors. The keywords are weighted according to their occurrence. The importance of a keyword depends on the number of times it appears in the content of an actor's documents.

$$Affinity = \frac{\sum_{i=1}^n \exists\{L_{X_i}, L_Y\}}{n}$$

The expression $\exists(L_{X_i}, L_Y)$ gives 1 if the keyword i from the keyword list of actor X is found in the keyword list of actor Y or 0 if does not exist. Therefore, if two actors have 5 of the 10 keywords in common, their affinity will be 0.5.

Having analysed the tests carried out, it was concluded that a study of the importance of the word within the whole social network was required. If there is a high incidence of the word in the whole social network, this does not determine a greater degree of affinity between two actors as it is present in the whole network. Alternatively, if the word is common to two actors but not frequently used in the social network, this indicates that the connecting link is due to the correlation of the semantic interests of both actors which the rest of the social network do not share. Therefore the initial calculation which weighted this factor has been modified and the following formula obtained:

$$Affinity = \frac{\sum_{i=1}^n \exists\{L_{X_i}, L_Y\} \cdot \frac{Occurrences(L_{X_i})}{GlobalOccurrences(L_{X_i})}}{n}$$

By this means, if actors X and Y coincide in 5 out of the 10 keywords, but these words are commonly used in the social network, the number of global occurrences will be very high, which will make the factor reduce the affinity between actors.

3. Visualization

Although it is not the main focal point of this research, it is essential that the environment for the presentation and navigation of the extracted social network is friendly and usable. An environment in which results can be easily obtained will facilitate future analysis and corresponding conclusions. Three functionalities have been designed for this purpose:

- Visualiser of the graph of an actor: this provides us with a vision of the social network, showing the actor and the different ties he shares with other actors. The keywords associated with the actor are also shown. The implementation of this function is based on the work of Moritz Stefaner [13]
- Tag cloud: the most relevant *keywords* of the whole social network are shown. This allows for the filtering by the given words with matching actors. These words are then used in subsequent filters carried out on the actors

- Navigation panel: this enables us to analyse the various levels of the social network and to consult the affinity coefficients of each actor as well as providing information on the performance of the system in its searches and data extraction.

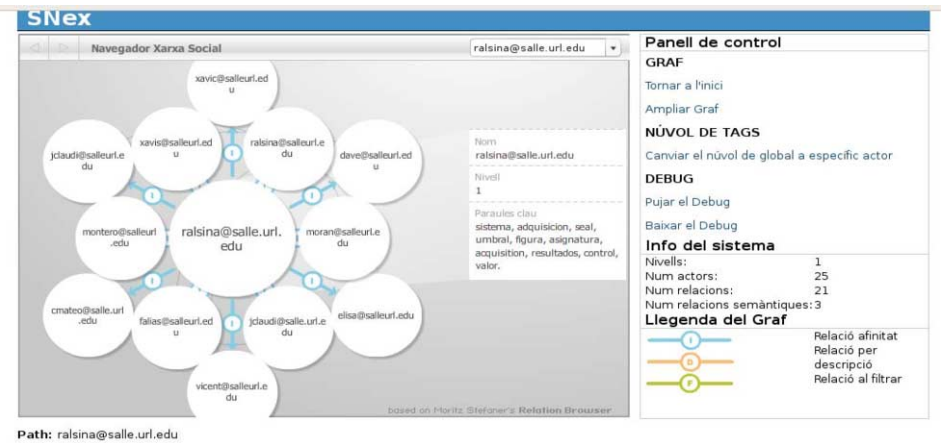


Figure 2. Visualisation of the social network

4. Results and Conclusions

Firstly, a comparative study of the response time of the two search engines on which SNex can currently consult information, Google and Yahoo, was carried out. The response time for 100 queries on the Yahoo engine was 85.620 seconds (0.86 s/consultation), while for Google it was 43.480 seconds. Here we can see how the response time of Google is much faster than that of Yahoo, almost half.

With reference to the method used to detect the language, tests on 81 PDF documents were carried out and the percentage of matches was 96%. In fact, on analysis of the three documents in which the language was wrongly identified, we saw how this error was due to a coding problem related to the PDF format. Therefore we can conclude that the detection method works correctly.

Turning to the reliability of the networks obtained by the automatic extraction process, the fact that the test environment consisted of queries made by research professors from our university centre should be underlined. The election of this environment permits us to verify the results obtained by consulting with those involved.

The first tests were carried out on known authors and the results obtained were absolutely correct. That is to say, the actors with a degree of co-authoring in the PDF documents were found. This meant that none of the actors were eliminated from the process given the fact that co-authoring implies a strong tie between actors.

Another of the tests carried out was realized with *ralsina*. This person was chosen due to her productivity in research articles. A level 1 network was initially created with 93 related actors. However, many of these actors had no real ties with the candidate. The reason for this was that co-authoring had not been penalised at this point and that sending lists from PDF documents had been classified as co-authors of an article. The results obtained were significant. From the 93 candidates, only 7 were actually actors with real ties. Applying the Overlap coefficient, 14 ties emerged, while when the

Cosinus coefficient was applied only 6 were found. The results obtained with cosinus are closer to the truth, if we consider that only one real tie was lost, while the Overlap coefficient produced 8 additional “unreal” actors related to the main actor.

Another noteworthy consideration is that the very nature of the semantic affinity formula is not commutative. For example, $Affinity(X,Y)$ does not necessarily have to be equal to $Affinity(Y,X)$. This has been proven in the tests. Although it is true that it might not seem logical at first, a series of tests demonstrated how the importance or strength of the tie between two actors does not give the same result when analysed from each of their different perspectives or roles within the social network. It may be the case that actor X belongs to a social network simply because he is linked to actor Y who forms part of the same network. In this case, the degree of affinity from actor X towards Y should be high. Nevertheless, the degree of affinity from actor Y towards X is not necessarily as relevant as in the previous relationship given that Y could be linked to other actors who are more important to him than actor X.

This application has been developed with the adoption of a vision-controller model which separates the layers of data, business logic and user interface. The SNex application has been developed using the PHP programming language and the MySQL database. With reference to system performance, the extraction system was extremely slow at first. To resolve this issue, *multithreading* was incorporated into the system so that several queries could be made on the search engines simultaneously. Response time on the search engines was reduced. The results were then stored in the memory cache and on hard disk to avoid redundant or repeated queries. This method resulted in a drastic time reduction. At present, it usually takes between 100 and 350 seconds to generate a level 1 social network.

5. Future Work

Focusing on SNex 1.0 and the results obtained, there are multiple future lines of research whose goal is to improving performance levels of the social network extraction system and to develop more precise methods to calculate affinity levels.

As far as web crawling is concerned, the possibility of testing the system with several additional search engines is gaining ground. The possibility of using the results obtained by the different search engines and assigning a weighted factor to each one is also being considered. Another aspect to take into account is how to improve the search techniques of the search engines.

The improvement strategies to work on the web mining stage are multiple. On one hand, proposals have been made to widen the scope of the type of information which is analysed. At present, only documents in PDF format are analysed but there are plans to study whether the addition of other types of document (html, postscript, text, etc.) would improve the quality of information and lead to better results in the social network.

In the area of extraction of semantic information, more complex algorithms are set to be included in this area in order to obtain semantic abstractions, such as the use of Wordnet at Princeton University.

Research and development into affinity calculations will continue and both coefficients and semantic descriptors will be employed. In this area, the investigative qualities of the researcher tend to be assessed and quantified by the number of references made by other authors in their articles.

Acknowledgements

The authors would like to thank Ingeniería i Arquitectura La Salle (Universitat Ramon Llull) for their encouragement and assistance, especially Lisa Kinnear for the linguistic reviews of the paper.

References

- [1] R.S. Wurman, "Angustia informativa", Prentice Hall, Argentina, 2002.
- [2] F. De la Rosa, R. Martínez, Sistemas de Inteligencia Web basados en Redes Sociales, Revista hispana para el análisis de redes sociales, 12(9), 2007.
- [3] H. Kautz, B. Selman, M. Shah, The hidden web, The AI Magazine, 18(2), 27-36, 1997.
- [4] P. Jaccard, Distribution de la flore alpine dans la bassin de transes et dans vuelques regions voisines. Bulletin de la societe Vaudoise des Sciences Naturelles, 241-272, 1901.
- [5] P. Mika, Flink: Semantic web technology for the extraction and analysis of social networks, Journal of Web Semantics, 3(2), 2005.
- [6] M. Hamasaki, Y. Matsuo, J. Mori, Polyphonet: an advanced social network extraction system from the web, Proceedings of WWW2006, 2006.
- [7] Yahoo! developer Network, [online] available: <http://www.developer.yahoo.com/search>, May 2008
- [8] Departamento de Computación, Universidad de Neuchatel, UniNE at CLEF, [online] available: <http://members.unine.ch/jacques.savoy/clef/>, June 2008.
- [9] D. Crystal, Diccionario de lingüística y fonética. Octaedro Editorial, 2000.
- [10] M.F. Porter, An algorithm for suffix stripping. Program, 130-137, 1980.
- [11] M.F. Porter, Snowball: a language for stemming algorithm. 1980.
- [12] M.R. Penabad, M.L. Moreda, Places, E. Vázquez, Algoritmo de stemming para el gallego. Procesamiento del Lenguaje Natural. 154-160, 2005.
- [13] M. Stefaner, Relation Browser, [online] available: <http://www.der-mo.net/relationBrowser/>, June 2008.

A NEGOTIATION STYLES RECOMMENDERS APPROACH BASED ON COMPUTATIONAL ECOLOGY

Josep Lluís de la Rosa, Gabriel Lopardo, Nicolás Hormazábal, and Miquel Montaner

{peplluís, glopardo, nicolash}@eia.udg.edu, miquel@isac.cat

Agents Research Lab - EASY Innova

Universitat de Girona (UdG) & Strategic Attention Management (SAM)

E17071 Girona (Catalonia)

Abstract This paper introduces a model of negotiation dynamics from the point of view of computational ecology. It inspires an ecosystem monitor, as well as a negotiation style recommender that is novel in state-of-the-art of recommender systems. The ecosystem monitor provides hints to the negotiation style recommender to achieve a higher stability of any instance of open negotiation environments in a digital business ecosystem.

Keywords: negotiating agents, recommender systems, computational ecologies, digital business ecosystems, negotiation styles

1. Introduction

Negotiation can be defined as an interaction of influences. Such interactions, for example, include the process of resolving disputes, agreeing upon courses of action, bargaining for individual or collective advantages, or crafting outcomes to satisfy various interests. Negotiation involves three basic elements: process, behavior, and substance. Process refers to how parties negotiate: the context of the negotiations, the parties to the negotiations, the tactics used by the parties, and the sequence and stages in which all of these play out. Behavior governs the relationships among these parties, the communication between them, and the styles they adopt. Substance refers to what the parties negotiate over: agenda, issues (positions, and – more helpfully – interests), options, and the agreement(s) reached at the end.

A hint in negotiations is to be prepared to walk, that is to be prepared to abort the negotiation (the exit strategy). If you are prepared to walk, this is real power in negotiations [9]. We will further develop this idea in this paper, and will formulate it as agent choosing deadlines and exit conditions in section 3. According to the literature ([1] and [3]), business negotiation can be classified into two types: auctions and negotiations in a narrow sense (which can be also considered reverse auctions). Auctions primarily focus on price negotiation and follow a clearly structured procedure. Compared to auctions, negotiations are not exclusively based on the competitive approach, but are based on a more unstructured dealing and bargaining negotiation process. Business negotiations are based on specific legal documents and follow a specific workflow, which can vary in the number of iterations necessary to reach an agreement [3].

A negotiation can start with publishing tender (by a tenderer) and inviting interested companies to provide offers (offerors). An unaccepted or missing answer ends the contract negotiation. Contract negotiation continues with a counter invitation to treat or a counter offer.

We will focus on the number of counter-offers in a negotiation, and will name them as “steps” in a negotiation, which can also be referred to as “interactions.” This paper deals with negotiation styles as a number of counter-offers. Reference [2] explains four types of opposite styles: cooperative vs. competitive and passive vs. active. Contemporary literature does not classify the number of steps in negotiations using the aforementioned negotiation styles. We will suggest a classification of panic vs. confidence, wherein a panicked negotiator settles immediately and a confident negotiator settles in the long run. By considering that the pace of negotiation differs from country to country, and that the pace of negotiations in the United States is faster than in most other cultures [9], we may infer that this behavior (being fast) must have an impact on the outcome of negotiations. Generally speaking, for successful negotiations, a shorter negotiation is worse for the offeror and better for the tenderer. Having very few negotiation steps spread out a very long time windows (the chinese style) is much worse for the offeror, and much better for the tenderer. In contrast, long negotiations, despite aiming for a better outcome, suffer from further risk of deadlocks that may break the negotiations, with the following risk of no outcome. In the long term, if one never settles, then there is no benefit at all. It seems that the optimal outcome is in the middle, that is, not immediately and not very long. Although some investigations ([3] [9]) have suggested saying “no” a final time when the deal seems close, whenever two agents apply this “say no, one more time” algorithm, then it is heuristically better if they have a different “say no, one more time” pattern. Thus, diversity in negotiation style could have a positive impact on populations of negotiating agents.

There is no negotiation style recommender in either state-of-the-art recommender systems [5], or in recommender agents on the Internet [13]. Therefore, there is an opportunity to create a new category of services devoted to the recommendation of the negotiation style for agents that negotiate in an open negotiation environment (ONE), in a similar fashion to those agents who deliver their opinion to other agents when they have to provide a recommendation to their users in [14].

Here, we are interested in the dynamics of an ONE within a digital business ecosystem (DBE), studied within a cluster of digital business ecosystems www.digital-ecosystems.org and in a project of the same name <http://one-project.eu>. This paper follows this structure: Section 2 presents a new negotiation style model and its dynamics, Section 3 describes a simple monitor and negotiation style recommender with examples of their functioning, and Section 4 contains the final remarks.

2. A Negotiation Model Inspired by Computational Ecologies

Perceptions and expectations concerning the current and future behavior of (eco)systems in many economic and social systems play an important role in determining the actions of an individual agent. References [9] and [10] closely investigated the dynamics of a model which captures the essential features of computational ecosystems, and analyzed these systems with computer simulations to

gain insight into the effects of time delays, cooperation, multiple resources, and heterogeneity. Reference [12] also investigated the effect of predictions on a model of co-evolution systems that incorporate many of the features of distributed resource allocation in systems comprised of many individual agents. They showed that system performance can be improved if agents correctly predict the current state of the system. Furthermore, reference [6] considered a procedure for controlling chaotic behavior in systems composed of interacting agents making decisions based on imperfect and delayed information. Their procedure used a reward mechanism, whereby the relative number of agents following effective strategies is increased at the expense of others. It is known as the Hogg-Huberman model (or simply as the Hogg model). References [8] and [15] have investigated the dynamical properties of a discrete-time Hogg model.

Reference [6] proposed a computational ecology model of the interaction of a large number of agents and their competition for resources. Agents in a computational environment appraise what computational power they will receive when they choose one resource over another. Unfortunately, they do not have instant access to the information of how much computational power is available at every source, and thus, the real computational power differs from the expected power, and worse, tends to be lower because of resource competition.

We propose a new model for negotiations, inspired from Hogg Model. We will refer to this model as the de la Rosa model for multiple agents' negotiation dynamics. This model consists of an interaction between a large number of agents and their competition for negotiating and settling deals with tenderers. The negotiating agents (named offeror agents, or simply offerors) appraise the wealth, benefit, profit, or income they may obtain after settling with tenderers, if they choose one tenderer instead of others. Although they may have instant access to tenderers' wealth information and information about the benefits of former negotiations with any of the negotiation agents, they do not know the real benefits (if any) available at the moment they set a deal. This is because many factors affect the progress of any negotiation, and thus, the actual benefits will diverge from what is expected. The benefits also tend to be lower than expected due to the considerable competition for the most healthy tenderers.

Let us demonstrate the de la Rosa model in reference to the dynamical behavior of a community of offerors negotiating with 2 tenderers within an ONE instance. This gives us two sources of business opportunities (the resources in business ecosystems) to which offerors must negotiate. We define $f_r(t)$ as the fraction of the offeror agents negotiating with tenderer r at time t . An interaction can occur in a two-tenderer ecosystem, wherein offeror agents re-evaluate the tenderers' wealth as they continue to negotiate with them, attempting to maximize the benefit of contracts from those tenderers. The offeror choice is modeled by the following equation:

$$\frac{df_r}{dt} = \alpha(\rho_r - f_r) \quad (1)$$

where α is the rate at which agents re-evaluate their tenderer choice (whether fruitful or not) and ρ is the probability that an agent will prefer Tenderer 1 over 2. Population of agents f_1 and f_2 are fractions of the offeror agents negotiating with tenderers 1 and 2, respectively.

The number of agents using the same fruitful tenderer as a resource for business contracts increases until too many agents are negotiating with this tenderer, while the payback by offeror diminishes because of the increasing competition, with the result

that prices decrease and thus benefits drop. Thus, the uncertainty (σ) is modeled as a typical deviation on performance to decide when tenderer 1 clearly drove an offeror agent to a more profitable contract (higher payoff) than tenderer 2. erf is the Gauss error function.

$$\rho = \frac{1}{2} \cdot \left(1 + erf \left(\frac{G_1(F(f_1)) - G_2(F(f_2))}{2\sigma} \right) \right) \tag{2}$$

When working with fast (no) negotiation, that is, closing the deal very quickly and receiving the profit of the contract as soon as possible, the offeror agents' choice of tenderer stabilizes after a transient period, as shown in Fig. 1 (left), which shows the stabilized population of agents negotiating with tenderer 1. When the settlement is postponed (delayed), the offeror agents have access to the contract's profit by k sample times ago, and the ecosystem shows a chaotic behavior, as shown with the same tenderer in Fig. 1 (middle).

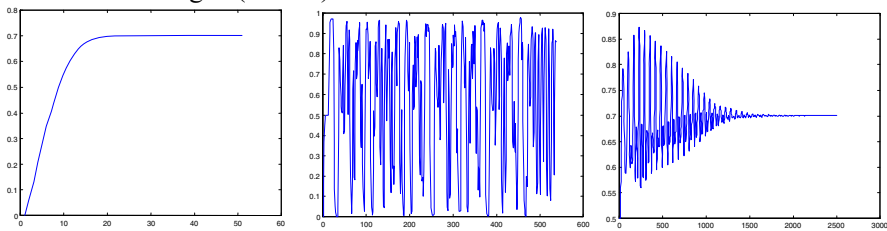


Fig. 1 (left) Offeror agents with no negotiation, and instant access to profit

Fig. 1 (middle) Offeror agents with delayed access to profit because of a long negotiation process

Fig. 1 (right) Stabilization of tenderer usage in a group of heterogeneous offeror agents

2.1. Similarities and Differences between the Hogg and de la Rosa models

The Hogg and de la Rosa models are similar, as the following map shows:

Table 1 Mapping among Hogg computational ecology and de la Rosa negotiation models

| Hogg Model (1) | de la Rosa Model (2) |
|---------------------------------|--|
| Computational Capacity of a CPU | Wealth of a Tenderer or its available money for tenders |
| Agents Appraisal of CPU | Offeror Agents Tenderer Wealth Appraisal |
| Delayed CPU information | Delayed settlement (of negotiations) |
| Delay: a number of time steps | Delay: a number of negotiation steps (indirect relation with time steps) |

The difference between the two models is that in the Hogg model, when an agent a makes a decision d_a by appraising in time t with s steps delayed information $F(t-sT)$, it obtains the payoff $G(t, d_a(F(t-sT)))$ immediately as computational capacity of the selected resource, while in the de la Rosa model, the agent makes a decision by using current information to appraise what payoff it will obtain in the future (after s negotiation steps) from the wealth of the selected resource.

It can be seen that both models are equivalent:

Being $G(t, d_a(F(t-sT)))$ the Hogg model

and $G(t+sT, d_a(F(t)))$ the de la Rosa model
 they are equal making $t'=t+sT$ (where model 2 turns $G(t', d_a(F(t'-sT)))$)

Therefore, some of the results obtained by Hogg can be mapped to the de la Rosa negotiation model. For example, Hogg, in his paper [6], shows how 40 heterogeneous types of offeror agents stabilize an ecosystem. In our work, each individual agent has a different delay to settle and access the profit/benefit. The delay, therefore, defines the negotiation style. This approach presents some interesting features:

- The entire system benefits from heterogeneity, allowing the whole ecosystem to stabilize the tenderee preference for negotiation. That is, the size of the offeror agents' community stabilizes around every tenderee.
- The stability of a system is determined by the behavior of a perturbation around equilibrium. Since heterogeneous systems are more stable than homogeneous systems [10], the important issue is to calculate how much diversity ("the mix") is needed to stabilize the ecosystem. In our case, the diversity is represented by the set of different delays that offeror agents can use when performing their negotiations.
- It presents an interesting and simple mechanism to deal with heterogeneity in multi-agent systems, by means of a reward mechanism that introduces competition among the offeror agents through the "negotiation style" recommender, which will be introduced in the following section.

3. A Digital Business Ecosystem Monitor and a Negotiation Style Recommender

A monitor measures the distribution of the offeror agents' population in terms of negotiation style. The monitor tries to measure the stability of the populations of offeror agents negotiating with the tenderees in the ecosystem. A recommender uses the information from the monitor to stabilize the digital business ecosystem by balancing the population of different types of offeror agents that negotiate with the tenderees. This is achieved by recommending changes in the offeror agents' negotiation style, affecting the number of steps to settle. The recommender suggests shifting by α steps the type s that defines the style of negotiation of every offeror agent towards another type, $s+\alpha$, where α is an integer.

Therefore, the monitor will take care of balancing the population of agents, conveniently categorized in negotiation styles parameterized by s negotiation steps. The idea is as follows:

- S is the number of negotiation styles.
- T_1 and T_2 are two different time windows (T).
- O is an instance of an ONE, a type of DBE.
- $f_s, f_r \subseteq O$ are subsets of negotiating offeror agents of type s and r
- $\varepsilon, \theta \in \mathbb{R}$.
- Monitor (O, S, T) measures the stability in offeror populations at an ONE O , with S types of agents, in a time window T . The higher stability, the better. Direct or indirect measures of stability include typical deviation, entropy, and the Shannon information measure.

The recommender will balance the offeror agents' population of every type S (negotiation style), and informs agents of in-use negotiation styles, such that agents

change from using a style with a “rather larger” population to using a style with a “rather small” population. The actions are as follows with the following conditions:

1. If $\text{Monitor}(f_s, S, T) < \text{Monitor}(f_r, S, T)$ then recommend to agents of type s a change/shift to type r
2. If $\text{Monitor}(O, S, T_1) < \varepsilon$ (unstable) then $S := S + 1$ (the number of negotiation styles is increased)
3. Any population of negotiation style s that stays small ($f_s < \theta$) during T_2 will disappear by making $S := S - 1$ (the number of negotiation styles is decreased), and the recommender tries to extinguish the population f_s by distributing/recommending the agents of this population to other populations with different negotiation styles.

3.1. Example with Two Tenderers and Three Negotiation Styles

The three negotiation styles are $s=0$ (always accept), $s=1$ (accept after at least one counteroffer), $s=2$ (say “no” one more time).

A and B are the two tenderers, and all three scenarios are run under the same conditions:

- Contracts with tenderer A double the benefit of those with tenderer B.
- Initial conditions $f^A = f^B$, that is, the same number of offerors are negotiating with tenderer A and tenderer B. f^A denotes the total number of offerors negotiating with A, of any type, that is $f^A = f_0^A + f_1^A + \dots$. Analogously, the total number of offerors of one type s is $f_s = f_s^A + \dots$. The total number of offeror agents is 20.
- $\alpha = 1$, $\rho = 1$, and no agents are allowed to type change/shift.
- A maximum of 1 agent from any population change tenderer at every step.

Scenario 1. There are 20 offerors of type 0, $f_0 = 20$ ($f_1 = 0$). This means that all agents immediately settle with a “no” counteroffer. The run through time stamps $t = 1, 2, 3, \dots, 14$ are:

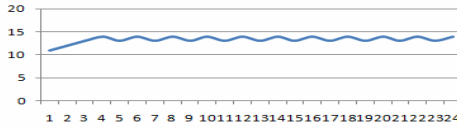
| Time t stamp | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| f^A | 10 | 11 | 12 | 13 | 14 | 13 | 14 | 13 | 14 | 13 | 14 | 13 | 14 | 13 |
| f^B | 10 | 9 | 8 | 7 | 6 | 7 | 6 | 7 | 6 | 7 | 6 | 7 | 6 | 7 |
| Profit from A | 1/5 | 1/5 | 1/6 | 1/6 | 1/7 | 1/6 | 1/7 | 1/6 | 1/7 | 1/6 | 1/7 | 1/6 | 1/7 | 1/6 |
| Profit from B | 0 | 1/9 | 1/8 | 1/7 | 1/6 | 1/7 | 1/6 | 1/7 | 1/6 | 1/7 | 1/6 | 1/7 | 1/6 | 1/7 |
| More Profit from | A | A | A | A | B | A | B | A | B | A | B | A | B | A |

One can see that there is a cycle of every two time stamps, that is, a frequency of 0.5.

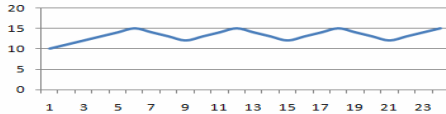
Scenario 2. There are 20 offerors of type 1, $f_0 = 0$, $f_1 = 20$. That is, all agents settle with delay 1, after 1 counteroffer. The run through time stamps $t = 1, \dots, 15$ are:

| Time t stamp | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|------------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| f^A | 10 | 10 | 11 | 12 | 13 | 14 | 15 | 14 | 13 | 12 | 13 | 14 | 15 | 14 | 13 |
| f^B | 10 | 10 | 9 | 8 | 7 | 6 | 5 | 6 | 7 | 8 | 7 | 6 | 5 | 6 | 7 |
| Profit from A | 1/5 | 1/5 | 1/5 | 1/6 | 1/6 | 1/7 | 1/7 | 1/7 | 1/6 | 1/6 | 1/6 | 1/7 | 1/7 | 1/7 | 1/6 |
| Profit from B | 0 | 0 | 1/9 | 1/8 | 1/7 | 1/6 | 1/5 | 1/6 | 1/7 | 1/8 | 1/7 | 1/6 | 1/5 | 1/6 | 1/7 |
| More Profit from | A | A | A | A | A | B | B | B | A | A | A | B | B | B | A |

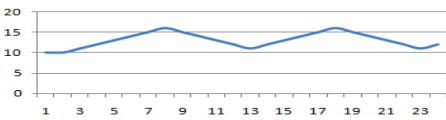
One can see that there is a cycle of every six time stamps, that is, a frequency of 0.1667. Experimenting with populations of different delay shows that $average(f^A) \approx 2 \cdot average(f^B)$. Other interesting measurements occur as well, as the following plots indicate:



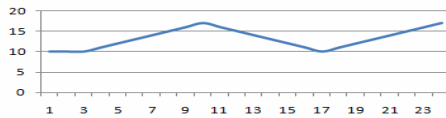
Type 0, no delay, freq=0.5, $\sigma^2=0.8$



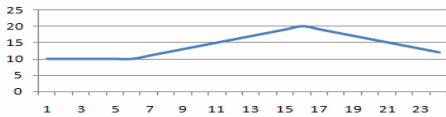
Type 1, 1 step, freq=0.17, $\sigma^2=1.1$



Type 2, 2 steps, freq=0.13, $\sigma^2=1.6$



Type 3, 3 steps, freq=0.07, $\sigma^2=2.1$



Type 7, 7 steps, freq= ?, $\sigma^2=2.7$

One can easily observe that as delay increases (the type of agents), variability increases (increasing σ^2), and the more difficult the decisions, driving toward chaotic behavior, as predicted by the Hogg model.

Scenario 3. This scenario introduces diversity into Scenario 2 by letting two agents be of type 0, and the remaining eighteen are of type 1 ($f_0 = 2$, $f_1 = 18$); $t = 1, \dots, 10$. This does not change the initial conditions of $f^A = f^B = 10$.

| Time t stamp | 1 | | 2 | | 3 | | 4 | | 5 | | 6 | | 7 | | 8 | | 9 | | 10 | |
|------------------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| | f_0 | f_1 | f_0 | f_1 | f_0 | f_1 | f_0 | f_1 | f_0 | f_1 | f_0 | f_1 | f_0 | f_1 | f_0 | f_1 | f_0 | f_1 | f_0 | f_1 |
| f^A | 1 | 9 | 2 | 9 | 2 | 10 | 2 | 11 | 2 | 12 | 1 | 13 | 0 | 12 | 1 | 11 | 2 | 12 | 1 | 13 |
| f^B | 1 | 9 | 0 | 9 | 0 | 8 | 0 | 7 | 0 | 6 | 1 | 5 | 2 | 6 | 1 | 7 | 0 | 6 | 1 | 5 |
| Profit from A | 1/5 | 1/5 | 1/5 | 1/5 | 1/6 | 1/6 | 1/6 | 1/6 | 1/7 | 1/7 | 1/7 | 1/7 | 1/6 | 1/6 | 1/6 | 1/6 | 1/7 | 1/7 | 1/7 | 1/7 |
| Profit from B | 0 | 0 | 1/9 | 1/9 | 1/8 | 1/8 | 1/7 | 1/7 | 1/6 | 1/6 | 1/6 | 1/6 | 1/8 | 1/8 | 1/8 | 1/8 | 1/6 | 1/6 | 1/6 | 1/6 |
| More Profit from | A | | A | | A | | A | | B | | B | | A | | A | | B | | B | |

As can be observed in the following plots (Fig. 2), the oscillation bands in Scenario 3 with the heterogeneous populations of two negotiation styles (0 and 1) is narrower than those of Scenario 2 (smaller typical deviation of 1.0 compared to 1.1 from Scenario 2), and with lower frequency than Scenario 1 (frequency of 0.25, twice as small as the 0.5 value from Scenario 1). We achieved this by simply introducing two agents of type 0!

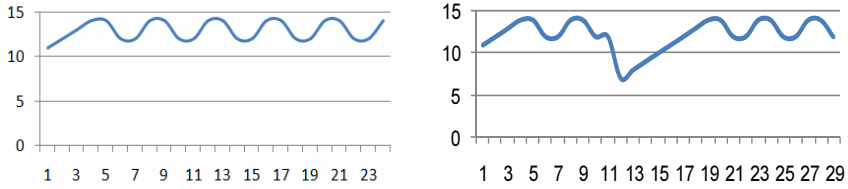


Fig. 2. Example of the stabilizing impact of a mixed population of agents, even with few agents of type 0 (on the left), wherein disturbances are rejected (on the right, disturbance at time stamp 13 is injected)

In summary, from the three scenarios, one can clearly observe how delayed settlements destabilize the negotiating ecosystem, and how stability can be reestablished by introducing diversity. These observations suggest a recommender will tell agents to shift their delays in settlements to gain global stability.

3.2. A Monitor and Negotiation Style Recommender Exemplified with 4 Populations

The recommendation strategy is to achieve a balance between negotiation style populations.

Primarily, the monitor will segment the S negotiation styles into N categories f'_i ; $i = 1$, such that several populations will fit in a category, that is, the category will contain several negotiation styles, and therefore $f'_i = f_{S_1} + \dots + f_S$. The number of categories depends on the granularity we use for the negotiation styles, with the limit $N \leq S$, and what complexity we can reach with the recommendation strategy. The criterion to classify is to be investigated in the future, but as a first instance, we may think of a uniform distribution of types, that is $\forall i, M_i = \cdot$. Let us take $N = 2$, $f'_1 = f_0 +$ as the low number of negotiation steps category, and $f'_2 = f_2 +$ as the high number of negotiation steps category. If we put the categories on a one-dimensional axis from a low to high number of steps, then the category will be the left negotiation styles and the category will be the right negotiation styles. In short, we call category the 'left' and the , the 'right'.

As previously described, the monitor will then continuously measure the number of steps that agents effectively use in their negotiations in every ONE, and will represent them in tables like Table 2 (left and right). The monitor will also calculate stability measures, such as the Shannon entropy H represented in Table 2, at every time stamp t . At each time stamp, the recommender will decide from what category agents must be shifted towards the other. In our example, it is very simple, that is, the recommender recommends that agents shift from left to right or vice versa. Since there are several categories, the recommendation will be much more complicated, and hence, will be studied in the future.

Table 2 (left) depicts an evolution of populations in our example experiments, wherein at every time stamp, one can see the shifts of agents from one type of negotiation style to another after measurement of their deal settling time. The shifts are spontaneous because agents vary the number of steps to settle due to various causes, including a tenderee change of conditions, difficulties in fulfilling the tenderee conditions, a change of the skills of the offerors, and simply because of their own decision as agents.

Table 2 (right) depicts the resulting recommendations at every time stamp. “L→R” indicates a recommendation that agents with lower types move to higher types, “L←R” the inverse, while “--” indicates that there is no recommendation. The result is that some agents try to modify, but few succeed in changing their negotiation style (it is not straightforward to change because the number of steps not only depends on offeror agents, but also on tenderees and competition). The evolution of Table 2 (right) shows the impact of recommendations on the spontaneous evolution of the population at every time stamp.

Table 2 (left), one can see a natural evolution of the population of four types of negotiation styles, while (right), shows the effect of the recommender on the balance of the populations

Populations only of Tenderee A

| Populations Left Right Entropy | | | | | | | | (*) Recommendation | Populations Left Right New Entropy | | | | | | | |
|--------------------------------|----------------|----------------|----------------|----------------|-----------------|-----------------|------|---|------------------------------------|----------------|----------------|----------------|----------------|-----------------|-----------------|------|
| t | f ₀ | f ₁ | f ₂ | f ₃ | f' ₁ | f' ₂ | H | | t | f ₀ | f ₁ | f ₂ | f ₃ | f' ₁ | f' ₂ | H' |
| 0 | 10 | 0 | 0 | 0 | 10 | 0 | 0,00 | L→R L→R -- L←R -- -- L←R L→R L→R L←R -- | 0 | 10 | 0 | 0 | 0 | 10 | 0 | 0,00 |
| 1 | 7 | 3 | 0 | 0 | 10 | 0 | 0,00 | | 1 | 4 | 3 | 3 | 0 | 7 | 3 | 0,88 |
| 2 | 5 | 3 | 2 | 0 | 8 | 2 | 0,72 | | 2 | 2 | 3 | 5 | 0 | 5 | 5 | 1,00 |
| 3 | 5 | 2 | 2 | 1 | 7 | 3 | 0,88 | | 3 | 2 | 2 | 5 | 1 | 4 | 6 | 0,97 |
| 4 | 4 | 3 | 2 | 1 | 7 | 3 | 0,88 | | 4 | 1 | 4 | 4 | 1 | 5 | 5 | 1,00 |
| 5 | 4 | 3 | 1 | 2 | 7 | 3 | 0,88 | | 5 | 1 | 4 | 3 | 2 | 5 | 5 | 1,00 |
| 6 | 3 | 3 | 1 | 3 | 6 | 4 | 0,97 | | 6 | 0 | 4 | 3 | 3 | 4 | 6 | 0,97 |
| 7 | 3 | 4 | 1 | 2 | 7 | 3 | 0,88 | | 7 | 1 | 5 | 3 | 1 | 6 | 4 | 0,97 |
| 8 | 4 | 4 | 1 | 1 | 8 | 2 | 0,72 | | 8 | 2 | 4 | 4 | 0 | 6 | 4 | 0,97 |
| 9 | 4 | 3 | 2 | 1 | 7 | 3 | 0,88 | | 9 | 2 | 2 | 6 | 0 | 4 | 6 | 0,97 |
| 10 | 3 | 3 | 2 | 2 | 6 | 4 | 0,97 | 10 | 1 | 4 | 4 | 1 | 5 | 5 | 1,00 | |
| average # 7,5 2,5 0,71 | | | | | | | | average # 5,5 4,5 0,88 | | | | | | | | |
| deviation 1,4 1,4 | | | | | | | | deviation 1,8 1,8 | | | | | | | | |

Entropy is represented as the H and H' columns at every time stamp t . The recommendation of shifts in the negotiation styles has produced sound increases of entropy, $H' > H$, which implies there is an improved population balance and higher diversity. According to the de la Rosa negotiation dynamics model, a higher diversity stabilizes the ONE, which improves the predictability and business opportunities of the whole digital business ecosystem, as shown in the example in 3.1.

4. Final Remarks

This investigation proposes a new category of recommenders, that is, the negotiation style recommender. They recommend a change in negotiation style to agents, which the agents execute by increasing or decreasing the number of steps towards settlements. This is a one-dimensional recommendation, and future work could focus on wider (multidimensional) negotiation style recommendations. Furthermore, as future work, the granularity in the classification of agents regarding their negotiation style, as well as the measurements obtained using business ecosystems monitors, need to be further investigated. Standard techniques, like collaborative and content-based filtering [5, 13], may be applied for negotiation styles, as well as to trust approaches for recommendation [13]. The originality of our approach is that we are looking for global behaviors that will increase opportunities at an individual level, although some recommendations may not be appropriate for an individual agent at certain moments.

Since these are only recommendations, agents may decide to follow them or not, or they can omit them because they follow the behavior of the tenderee (he is the boss!).

The result is a recommender schema that is conceptually simple, yet powerful, for large populations (thousands, billions, ...) of agents, and may work with low computational complexity. This simplicity comes from the de la Rosa negotiation dynamics model, inspired by a former Hogg model of computational ecologies.

The scenarios show that a higher diversity of offeror agents stabilize the ecosystem of negotiating agents, which is good according to the de la Rosa model. Experiments have demonstrated how the monitor tracks homogeneous populations of offerors (those with lower entropies) and lets the recommender introduce a diversity of negotiation styles (then the populations will have higher entropies). Future versions of the recommender will reward agents for their change of negotiation style, perhaps in the same way governments use taxes to introduce changes in economic behavior.

ACKNOWLEDGEMENTS

The research is partially funded by the European Union projects 34744 ONE: Open Negotiation Environment, FP6-2005-IST-5, and N° 216746 PReservation Organizations using Tools in AAgent Environments (PROTAGE), FP7-2007- ICT

REFERENCES

1. Bichler, M.; Kersten, G.; Strecker St. (2003). Towards a Structured Design of Electronic Negotiations. In: Group Decision and Negotiation, Vol. 12, No. 4, pp. 311-335.
2. Blake, R.R. & Mouton, J.S. (1969) Building a Dynamic Corporation through Grid Organization Development. Addison-Wesley
3. de Moore and Weigand (2004) Business Negotiation Support: Theory and Practice, International Negotiation, 9(1):31-57, 2004.
4. Gisler, M., Stanoevska-Slabeva, K., Greunz, M. (2000) Legal Aspects of Electronic Contracts, Proceedings of the Workshop of Infrastructures for Dynamic Business-toBusiness Service outsourcing (IDSO'00).
5. Gediminas Adomavicius, Member, IEEE, and Alexander Tuzhilin, Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions, IEEE Transactions on Knowledge and Data Engineering, Vol. 17, no. 6, June 2005
6. Hogg, T.; Huberman, B.A. (1991), Controlling chaos in distributed systems, IEEE Trans. Syst. Man Cybernet. 21 (6) 1325.
7. Huberman, B.A.; Hogg, T. (1988) in: B.A. Huberman (Ed.), The Ecology of Computation, North-Holland, Amsterdam, 1988, p. 77.
8. Inoue, M.; Tanaka, T.; Takagi, N.; Shibata, J. (2002) Dynamical behavior of a discrete time Hogg-Huberman model with three resources, Physica A 312: 627.
9. Karras C.L., Give and Take: The Complete Guide to Negotiating Strategies and Tactics, Harpercollins, : June 1993, ISBN: 0887306063
10. Kephart, J. O. Hogg T., and Huberman B. A. Dynamics of computational ecosystems. Physical Review A, 40:404-421, (1989a).
11. Kephart, J.O.; Hogg, T.; Huberman, B.A. (1989b) in: M.N. Huhns (Ed.) Distributed Artificial Intelligence, Vol. 2, Kaufman, Los Altos, CA, 1989.
12. Kephart, J.O.; Hogg, T.; Huberman, B.A. (1990) Physica D 42: 48.
13. Montaner M.; López B; de la Rosa J. Ll, A Taxonomy of Recommender Agents on the Internet, ISSN 0269-2821, Artificial Intelligence Review, Vol: 19, pp: 285-330, Kluwer Academic Publishers, Berlin, 2003
14. J. L. de la Rosa, M. Montaner, J.M. López, Opinion-Based Filtering (Trust Aware Recommender Agents), 17th European Conference on Artificial Intelligence ECAI 2006, Workshop on Recommender Systems, pp: 84-87, Trento, Italy, August 26-29, 2006
15. Ushio, T.; Inamori, T. (1997) Proceedings of the 36th IEEE CDC, San Diego, CA, 1997, p. 389.

CABRO: Winner Determination Algorithm for Single-unit Combinatorial Auctions

Víctor MUÑOZ and Javier MURILLO
{vmunozs, jmurillo}@eia.udg.es
University of Girona

Abstract. In this paper we present CABRO, an algorithm for solving the winner determination problem related to single-unit combinatorial auctions. The algorithm is divided in three main phases. The first phase is a pre-processing step with some reduction techniques. The second phase calculates an upper and a lower bound based on a linear programming relaxation in order to delete more bids. Finally, the third phase is a branch and bound depth first search where the linear programming relaxation is used as upper bounding and sorting strategy. Experiments against specific solvers like CASS and general purpose MIP solvers as GLPK and CPLEX show that CABRO is in average the fastest free solver (CPLEX not included), and in some instances drastically faster than any other.

Keywords. winner determination problem, combinatorial auctions

Introduction

Research in combinatorial auctions has grown rapidly in the recent years. This kind of auctions allow bidders to place bids on combinations (also named bundles, collections or packages) of items rather than just on individual items. This fact allows bidders to better express their interests as well as other restrictions such as complementarity or substitutability [9,4]. In a single-unit combinatorial auction the seller (auctioneer) is faced with a set of bids with different prices, and his aim is to select the best subset of them (maximizing the sum of its prices), the *winners*, so that there does not exist any pair of them sharing any item. The problem of selecting the winners in an auction is known as the *Winner Determination Problem* (WDP) and is particularly difficult in combinatorial auctions as it is equivalent to the weighted set-packing problem and the maximum weighted clique problem and therefore *NP-hard* [8,7]. Furthermore, it has been demonstrated that the WDP cannot even be approximated to a ratio of n^{1-e} (any constant factor) in polynomial time, unless $P = NP$ [9].

Since 1998 there has been a surge of research on designing efficient algorithms for the WDP (see [3,7] for a more extended survey). Given that the problem is *NP-Hard* in the strong sense, any optimal algorithm will be slow on some problem instances. However, in practice, modern search algorithms can optimally solve the WDP in a large variety of practical cases. There exist typically two different ways of solving it. On one hand

there exist specific algorithms that have been created exclusively for this purpose, such as CASS [4] and CABOB [10]. On the other hand, the WDP can be modeled as a mixed integer linear problem (MIP) and solved using a generic MIP solver. Due to the efficiency of actual MIP solvers like GLPK (free) and specially CPLEX (commercial), the research community has nowadays mostly converged towards using MIP solvers as the default approach for solving the WDP. There also exist sub-optimal algorithms for solving the winner determination problem that find quick solutions to combinatorial auctions [11,12]. However we will focus only on optimal solutions.

An interesting thing to be noted about the modeling of the WDP as a MIP is that if bids were defined in such a way that they could be accepted partially, the problem would become a linear program (LP) which, unlike MIP, can be solved in polynomial time. We have kept this idea in mind to design a new algorithm, CABRO, which combines LP, search and several reduction techniques to obtain better results than other solvers, even CPLEX in some particular instances.

1. Notation

Here we introduce a few notation that is going to be used through this paper. In a single-unit combinatorial auction the auctioneer receives a set of bids $B = \{b_1, \dots, b_n\}$, each of them composed by a price $p(b_i)$ and a subset of items $g(b_i)$ of size $n(b_i)$ (such that $n(b_i) = |g(b_i)|$). The complete set of items is $I = \{it_1, \dots, it_m\}$.

Useful relations between bids include $b(it_i)$ as the set of bids that contain the item it_i , and $C(b_i)$ as the set of bids compatible with bid b_i (i.e. the set of bids that do not contain any item in $g(b_i)$). Additionally, $C(b_i, b_j)$ and $\neg C(b_i, b_j)$ represent whether bids b_i and b_j are compatible or incompatible.

2. The Algorithm

CABRO (Combinatorial Auction BRanch and bound Optimizer) is mainly a branch and bound depth-first search algorithm with a specially significative procedure to reduce the size of the input problem. The algorithm is divided in three main phases:

- The first phase performs a fast preprocessing (polynomial time) with the aim of removing as many bids as possible. Bids removed in this phase may be either bids that are surely not in the optimal solution, or bids that surely are.
- The second phase consists in calculating upper and lower bounds for each bid. The upper bound of a bid is computed by formulating a relaxed linear programming problem (LP), while the lower bound is computed generating a solution quickly. This phase may also remove a notable amount of bids.
- The third phase completes the problem by means of search, concretely a branch and bound depth first search. In this phase the two previous phases are used also as heuristic and for pruning.

In some instances it is not necessary to execute all the three phases of the algorithm, for example when the optimal solution is already found before the search phase (which happens more frequently than expected). The algorithm is able to end prematurely either

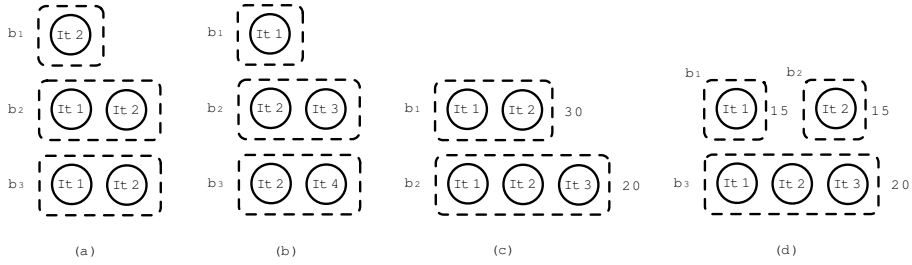


Figure 1. Examples of (a) dominated item (it_1), (b) solution bid (b_1), (c) dominated and (d) 2-dominated bids.

when all of the bids have been removed or when at some point of the execution the global lower bound reaches the global upper bound.

This algorithm also provides anytime performance, giving the possibility to be stopped at any time during the execution and providing the best solution found so far. In the following sections each of the three phases of the algorithm are explained in detail.

2.1. First phase: Pre-processing

This phase uses fast algorithms (with polynomial-time complexity) to reduce the size of the problem by deleting bids and items that either cannot be present at the optimal solution or that surely belong to it. This phase consists of 8 separate strategies (steps), each of them using a different criteria to remove either bids or items.

- Step 1: Bids with null compatibility.** In this step all the bids that do not have any compatible bid are deleted, except for the bid with the highest price b_h . These bids are surely not in the optimal solution since the maximum benefit of a solution containing any of them would be its own price, yet it still does not surpass the price of the bid b_h .
- Step 2: Dependent items.** Items give information about incompatible bids. Still in some cases the information given by an item is already included into another's: the item is *dependent*. Then, the former can be removed without any loss of information. Hence, this step deletes (leaves out of consideration) dependent items. More formally, for each pair of items (it_1 , it_2) such that $b(it_1) \subseteq b(it_2)$, it_1 may be deleted from the problem since the information given by it_1 is redundant. Figure 1 (a) shows an example of this situation; here item it_1 can be deleted given that the information given by it_1 ($\neg C(b_2, b_3)$) is already included in the information given by it_2 ($\neg C(b_1, b_2)$, $\neg C(b_2, b_3)$ and $\neg C(b_1, b_3)$).
- Step 3: Bids of the solution.** In some instances there may exist bids such that all of its items are unique (the bid is the only one containing them), and therefore the bid does not have any incompatible bid. In such situations the bid is surely part of the optimal solution. This step finds all the bids complying with this condition, adding them to the optimal solution and being removed from the remaining set of bids. Figure 1 (b) shows an example of this situation, where bid b_1 is added to the optimal solution

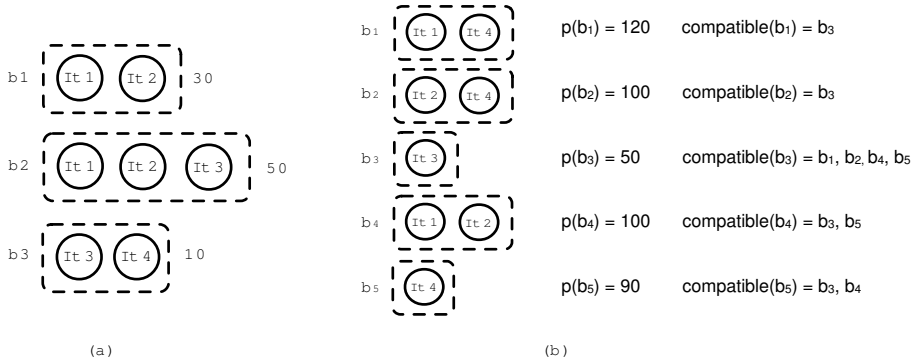


Figure 2. Left: Example of pseudo-dominated bid (b_1 is pseudo-dominated). Right: Example of compatibility-dominated bid (b_2 is compatibility-dominated by b_1).

given that its item it_1 is unique.

- Step 4: Dominated bids.** This is the same pre-processing step that CASS [4] and CABOB [10] perform: the elimination of dominated bids. A bid is dominated by another when its set of items includes another bid's items and its price is lower. More formally, for each pair of bids (b_i, b_j) where $g(b_i) \subseteq g(b_j)$ and $p(b_i) \geq p(b_j)$, b_j may be removed as it is never preferable to b_i . Figure 1 (c) shows an example of a dominated bid (b_1 dominates b_2).
- Step 5: 2-Dominated bids.** This is an extension of the previous technique (also noticed in [9]), checking whether a bid is dominated by a pair of bids. In some cases a bid is not dominated by any single bid separately, but the union of two bids together (joining items and adding prices) may dominate it. Figure 1 (d) shows an example of a 2-dominated bid (the union of b_1 and b_2 dominates b_3). This step can be easily generalized to check n -dominated bids. However, for many reasonable distributions, the probability of a bid being dominated by n bids is very low for higher values of n , still requiring much more processing (finding all subsets of size n), so this generalization is not useful at all for $n > 2$.
- Step 6: Pseudo-dominated bids.** This step is an even more complex generalization of the dominating techniques. Here we deal again with pairs of bids (b_i, b_j) such that not all of the items in b_i are contained in b_j , but there is one single item it_k not included. In this situation the bid b_i can be removed only if adding to its price the price of its best (highest price) compatible bid containing item it_k is not higher than the price of the bid b_j . In such a situation b_j is always preferable to b_i even when taking b_i together with its best compatible bid; therefore b_i does definitely not belong to the optimal solution and might be removed. Figure 2 (a) illustrates this situation: here b_2 pseudo-dominates b_1 since its price (50) is higher than the sum of bid b_1 's price (30) plus the price of its best compatible bid containing the item it_3 , in this case b_3 (10), therefore b_1 can be removed.

- **Step 7: Upper and lower bound values.** In this step, fast upper and a lower bounds are assigned to each bid with the aim of deleting bids with its upper bound lower than a *global lower bound* (GLB)¹, since they cannot improve the best solution already found.

The upper bound u of a bid b_x is calculated according to Equation 1 where $C'(b_x, it_k)$ is the set of compatible bids of b_x including item it_k . Roughly, it computes the upper bound of a bid b_i by adding to its price the best possible prices of the bids containing the items not included in $g(b_i)$.

After that, the lower bound of the bids is then calculated constructing a solution of a bid by iteratively attempting to add all of its compatible bids to the solution. Its compatible bids are ordered in descending order according to the upper bound previously calculated. All the solutions obtained with this algorithm are valid solutions and update the GLB accordingly. Note that GLB actually stores the best solution to the problem found so far (although it may not be the optimal one), therefore it can be returned immediately if the user decides to stop de execution, thus providing anytime performance.

$$u(b_x) = p(b_x) + \sum_{\forall i \notin g(b_x)} \max_{\forall j \in C'(b_x, it_k)} \frac{p(b_j)}{n(b_j)} \quad (1)$$

- **Step 8: Compatibility-Dominated bids.** This step is another generalization of dominated bids. A bid b_i is compatibility-dominated by another bid b_j if the set of compatible bids of b_i is a subset of the set of compatible bids of b_j and its price is lower. More formally, for each pair of bids (b_i, b_j) where $C(b_i) \subseteq C(b_j)$ and $p(b_i) \geq p(b_j)$, b_j may be removed as it is never preferable to b_i . Figure 2 (b) shows an example where b_2 is not dominated by b_1 but it is compatibility-dominated.

Once all of these steps have been executed, since the problem has changed, it may be the case that some bids and items previously undeleted can now be removed. For example the deletion of a bid may cause the appearance of dominated items and vice-versa. Therefore phase 1 is repeated until it does not remove any more bid or item.

2.2. Second phase: Upper and Lower Bounding

In the second phase, the algorithm calculates improved upper and lower bounds for each bid. In order to compute the upper bound for a given bid b_i , a relaxed linear programming (LP) problem is formulated. This relaxed formulation defines the bids in such a way that they can be accepted partially (a real number in the interval $[0, 1]$), therefore it can be solved using the well-known simplex algorithm [2], which solves most of the instances in polynomial-time. The relaxed version does not contains neither the current bid b_i nor none of the bids with items included in b_i (i.e. its incompatible bids). Adding the price of the bid b_i to the solution of the relaxed LP problem gives a new upper bound that is usually much more precise than the one obtained in step 7 of phase 1.

This step firstly performs an ordering of the bids according to the upper bound value calculated in step 7 of phase 1 in ascending order. Then the process of calculating new

¹The global lower bound (GLB) is the best (maximum) lower bound found, associated to a valid solution.

upper bounds using the simplex method starts with the bid with the lower upper bound, and each time a bid's upper bound is lower than the GLB, it is deleted, thus decreasing the size of the subsequent bids' simplex. Note that the chosen ordering, beginning with the "worst" bids, may seem inappropriate, but this is in fact a good strategy since the worst bids' upper bounds are usually much faster to compute than the "best", hence we quickly obtain accurate upper and lower bounds that may allow to remove lots of bids rapidly, thus decreasing the size of the problem and making "best" bids also faster to be computed. This fact has been verified experimentally.

Regarding the lower bound for each bid b_i , it is computed using the values returned by the LP solver, and updates the GLB accordingly. The solution is constructed by firstly considering any value greater than 0.5 to be actually 1; that is, part of the (partial) solution. This assumption is not inconsistent (it does not produce solutions containing incompatible bids) because compatible bids are restricted to sum at most 1, therefore two incompatible bids cannot have both values larger than 0.5. After that, the remaining bids (with values smaller or equal to 0.5) are attempted to be put into the solution in descending order. Of course if the solution of the LP was integer this process is not required, as it is the optimal solution for that bid.

2.3. Third phase: Search

The third phase (*iCabro*) performs a branch-and-bound depth-first search with the remaining bids of the previous phases (L). The full algorithm can be seen in Figure 3. The value of the best solution found so far (GLB) is stored in the global variable $bSolution$. Initially $bSolution=0$, and the search starts by calling $iCabro(L,0)$.

```

1      procedure iCabro( $L, cSolution$ )
2      for each element  $b$  of  $L$ 
3           $L2 \leftarrow L \cap compatible(b)$ 
4           $cSolution2 \leftarrow cSolution \cup b$ 
5           $LPSol \leftarrow simplex(cSolution2)$ 
6          if  $LPSol$  is integer then
7               $cSolution2 \leftarrow cSolution2 \cup LPSol$ 
8               $L2 \leftarrow \emptyset$ 
9          end-if
10         if  $v(LPSol) > v(bSolution)$  then
11             if  $v(cSolution2) > v(bSolution)$  then
12                  $bSolution \leftarrow cSolution2$ 
13             end-if
14             if  $L2$  is not empty then
15                  $sort(L2)$ 
16                  $iCabro(L2, cSolution2)$ 
17             end-if
18         end-if
19     end-for
20 end-procedure

```

Figure 3. Pseudo-code algorithm of *iCabro* procedure

The *iCabro* procedure processes the incoming list of bids L performing the following steps:

- The algorithm begins getting the first bid b of the list L (recall that L is sorted according to the upper bound computed in phase 2). A new list $L2$ is created as the intersection between L and $C(b)$ (compatible bids of b). In deeper nodes (as it is a recursive function) the set $L2$ represents the compatible bids with the current solution.
- After that, the algorithm formulates and solves the Linear Programming (LP) problem related to the current solution. If the result of the LP problem is integer then the algorithm finishes (prunes) the current branch, as the optimal solution of the branch has been found.
- At line 10 the algorithm verifies if the upper bound of the current solution is greater than the GLB (the best solution found so far). If this is the case the search continues through this branch updating the best current solution if necessary. Otherwise, the branch is pruned.
- At line 14 the algorithm verifies that the $L2$ set is not empty, given that if it is empty then it means that the current solution does not have any more compatible bids and consequently the branch is finished. Alternatively, if this condition does not apply, then the following action is to sort the list $L2$ according to the upper bound of each bid, in order to perform a recursive call to *iCabro* with the list $L2$.

3. Results

To evaluate the CABRO algorithm we have compared it against both specific algorithms and general purpose MIP solvers. We have chosen CASS for the specific solver instead of CABOB because although their authors claim that it outperforms CASS, there is no implementation of it available publicly. For the MIP solver, both GLPK (free) and CPLEX 10.1 (commercial) have been tested.

Test examples have been generated using the popular benchmark for combinatorial auctions CATS (Combinatorial Auctions Test Suite) [6], which creates realistic auction instances. Since its first release in 2000, CATS has become the standard benchmark to evaluate and compare WDP algorithms [10,7]. The CATS suite generates instances following five real-world situations and seven previously published distributions by different authors (called legacy). Given a required number of goods and bids, all the distributions select which goods to include in each bid uniformly at randomly without replacement.

For most of the five real-world distributions a graph is generated representing adjacency relationships between goods, and it is used to derive complementarity properties between goods and substitutability properties for bids. Some of the real-world situations concern complementarity based on adjacency in (physical or conceptual) space, while the remaining concern complementarity based on correlation time. The characteristics of each distribution are as follows:

- Paths (PATHS). This distribution models shipping, rail and bandwidth auctions. Goods are represented as edges in a nearly planar graph, with agents submitting a set of bids for paths connecting two nodes.
- Arbitrary (ARB). In this distribution the planarity assumption is relaxed from the previous one in order to model arbitrary complementarities between discrete goods such as electronics parts or collectables.

- Matching (MAT). This distribution concerns the matching of time-slots for a fixed number of different goods; this case applies to airline take-off and landing rights auctions.
- Scheduling (SCH). This distribution generates bids for a distributed job-shop scheduling domain, and also its application to power generation auctions.

The seven legacy distributions are the following:

- L1, the *Random* distribution from [9], chooses a number of items uniformly at random from $[1, m]$, and assigns the bid a price drawn uniformly from $[0, 1]$.
- L2, the *Weighted Random* distribution from [9], chooses a number of items g uniformly at random from $[1, m]$ and assigns a price drawn uniformly from $[0, g]$.
- L3, the *Uniform* distribution from [9], sets the number of items to some constant c and draws the price offer from $[0, 1]$.
- L4, the *Decay* distribution from [9] starts with a bundle size of 1, and increments the bundle size until a uniform random drawn from $[0, 1]$ exceeds a parameter α .
- L5, the *Normal* distribution from [12], draws both the number of items and the price offer from normal distributions.
- L6, the *Exponential* distribution from [4], requests g items with probability $C_{e^{-g/q}}$, and assigns a price offer drawn uniformly at random from $[0.5g, 1.5g]$.
- L7, the *Binomial* distribution from [4], gives each item an independent probability of p of being included in a bundle, and assigns a price offer drawn uniformly at random from $[0.5g, 1.5g]$ where g is the number of items selected.

We have also created a new distribution called transports (TRANS) based on a real problem: the road transportation problem. The problem roughly consists of finding the best assignment of available drivers to a set of requested services given a cost function and subject to a set of constraints (see [1] for more details). To model this problem as an auction the bids represent journeys (a set of services) associated with a driver, therefore its items represent the services performed as well as the driver used. Note that the original problem consists in minimizing the final cost of doing all the services, while an auction is concerned on maximizing. Therefore, the costs associated to the bids are appropriately transformed so that the maximized solution corresponds to the real (minimizing) solution.

We have generated 100 instances of each distribution with different amounts of bids and items. Each instance has been solved using CABRO, CASS, GLPK 4.9 and CPLEX 10.1 with a timeout of 300 seconds. The first three methods have been run in a 2.4GHz Pentium IV with 2Gb of RAM running under Windows XP SP2, while CPLEX has been run on a 3.2GHz Dual-Core Intel Xeon 5060 machine with 2 Gb of RAM running under GNU/Linux 2.6.

Figure 4 (left) shows the average execution time (in seconds) required for each method to solve all the instances of all the distributions. Here we can observe that CPLEX is in average the fastest solver since it solves all the instances (1167 auctions) in considerably less time than the other solvers. Recall that the machine used for CPLEX is considerably faster than the one used for the others; however, we believe that the results on equal machines would not change significantly. Yet CABRO spends less time than the free solvers GLPK and CASS.

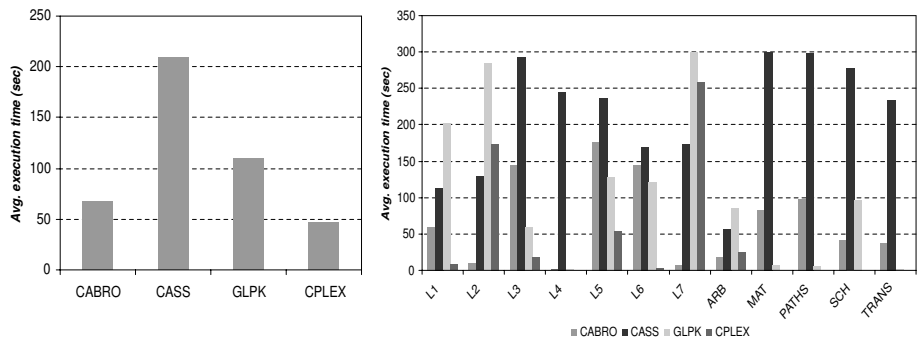


Figure 4. Left: Global comparative. Right: Comparative over distributions.

Figure 4 (right) shows the results in each of the distributions comparing the average time required (in seconds) to solve all the instances of each distribution with the four methods. Here we can observe that in two distributions (L2 and L7) CABRO is clearly the best algorithm and in other one (ARB) is also the best solver but CPLEX is very close. In the rest of distributions CPLEX is the best. Regarding the free solvers, GLPK is clearly the best solver in L3, L5, TRANS, MAT and PATHS while CABRO is clearly the best in L1, L2,L7, ARB and SCH. CASS is only rather competitive in L1, L2, L7 and ARB distributions.

| | CABRO | | | CASS | | | GLPK | | | CPLEX | | |
|--------------|----------|----------|-------|----------|----------|------|----------|----------|-------|----------|----------|-------|
| | <i>F</i> | $\neg F$ | % | <i>F</i> | $\neg F$ | % | <i>F</i> | $\neg F$ | % | <i>F</i> | $\neg F$ | % |
| <i>L1</i> | 80 | 15 | 84.2 | 67 | 28 | 70.5 | 39 | 56 | 41.1 | 95 | 0 | 100.0 |
| <i>L2</i> | 100 | 0 | 100.0 | 90 | 10 | 90.0 | 7 | 93 | 7.0 | 50 | 50 | 50.0 |
| <i>L3</i> | 54 | 46 | 54.0 | 3 | 97 | 3.0 | 84 | 16 | 84.0 | 98 | 2 | 98.0 |
| <i>L4</i> | 100 | 0 | 100.0 | 22 | 78 | 22.0 | 100 | 0 | 100.0 | 100 | 0 | 100.0 |
| <i>L5</i> | 44 | 56 | 44.0 | 23 | 77 | 23.0 | 61 | 39 | 61.0 | 90 | 10 | 90.0 |
| <i>L6</i> | 53 | 47 | 53.0 | 46 | 54 | 46.0 | 70 | 30 | 70.0 | 100 | 0 | 100.0 |
| <i>L7</i> | 100 | 0 | 100.0 | 68 | 32 | 68.0 | 0 | 100 | 0.0 | 15 | 85 | 15.0 |
| <i>ARB</i> | 96 | 4 | 96.0 | 86 | 14 | 86.0 | 81 | 19 | 81.0 | 99 | 1 | 99.0 |
| <i>MAT</i> | 81 | 19 | 81.0 | 0 | 100 | 0.0 | 100 | 0 | 100.0 | 100 | 0 | 100.0 |
| <i>PATHS</i> | 55 | 17 | 76.4 | 1 | 71 | 1.4 | 72 | 0 | 100.0 | 72 | 0 | 100.0 |
| <i>SCH</i> | 98 | 2 | 98.0 | 9 | 91 | 9.0 | 84 | 16 | 84.0 | 100 | 0 | 100.0 |
| <i>TRANS</i> | 94 | 6 | 94.0 | 24 | 76 | 24.0 | 100 | 0 | 100.0 | 100 | 0 | 100.0 |
| <i>TOTAL</i> | 955 | 212 | 81.8 | 439 | 728 | 37.6 | 798 | 369 | 68.4 | 1019 | 148 | 87.3 |

Table 1. Finished auctions (*F*), not finished auctions ($\neg F$) and percentage of finished auctions (%) before the timeout.

Table 1 shows the number of auctions finished (*F*), the number of auctions not finished ($\neg F$) and the percentage of finished auctions (%) before the timeout, for each method and each distribution. The results are similar to the execution time results, with CPLEX being the best method in absolute results, as it solves up to 1019 instances (87%). However, there is not any method that can be claimed to be the best, since it depends on the kind of data that the auction is processing. Particularly, CABRO performs better for

the weighted random and binomial distributions, solving 100% of the instances, while CPLEX only solves 15% in L7 and 50% in L2.

4. Conclusions

In this paper an algorithm for solving combinatorial auction problems has been presented. It uses many reduction techniques, together with an heuristic function based on linear programming techniques that provides more pruning. We have compared its performance with other existing algorithms obtaining encouraging results, particularly for weighted random and binomial distributions. In the future, we plan to hugely improve the algorithm with new reduction strategies as well as integrate it in the search phase. We also plan to improve the upper bound function used in the first phase and to test other sorting criteria to obtain better lower bounds. Also, we would focus on understanding the different characteristics of the domains and its influence in the solution time, and theoretically characterize domains where CABRO outperforms CPLEX and work in the domains where it does not. We expect that these changes would significantly improve the algorithm performance. Another interesting point would be to extend this algorithm to deal also with multi-unit combinatorial auctions, as there are not many specific algorithms for this kind of auctions. Finally, we will study another criteria to evaluate the algorithm as for example the anytime behavior and the memory consumption, as it is known to be a drawback of MIP solvers.

References

- [1] Javier Murillo and Beatriz Lopez, 'An empirical study of planning and scheduling interactions in the road passenger transportation domain', *Proceedings of PlanSIG 2006*, 2006., 129–136, (2006).
- [2] George B. Dantzig, 'The simplex method', *RAND Corp*, (1956).
- [3] Sven de Vries and Rakesh V. Vohra, 'Combinatorial auctions: A survey', *INFORMS Journal on Computing*, (3), 284–309, (2003).
- [4] Yuzo Fujishima, Kevin Leyton-Brown, and Yoav Shoham, 'Taming the computational complexity of combinatorial auctions: Optimal and approximate approaches', in *International Joint Conferences on Artificial Intelligence (IJCAI)*, pp. 548–553, (1999).
- [5] Holger H. Hoos and Craig Boutilier, 'Solving combinatorial auctions using stochastic local search', in *AAAI/IAAI*, pp. 22–29, (2000).
- [6] Kevin Leyton-Brown, Mark Pearson, and Yoav Shoham, 'Towards a universal test suite for combinatorial auction algorithms', in *ACM Conference on Electronic Commerce*, pp. 66–76, (2000).
- [7] Yoav Shoham Peter Cramton and Richard Steinberg, *Combinatorial Auctions*, MIT Press, 2006.
- [8] M. H. Rothkopf, A. Pekec, and R. M. Harstad, 'Computationally manageable combinatorial auctions', Technical Report 95-09, (19, 1995).
- [9] Tuomas Sandholm, 'Algorithm for optimal winner determination in combinatorial auctions', *Artificial Intelligence*, **135**(1-2), 1–54, (2002).
- [10] Tuomas Sandholm, Subhash Suri, Andrew Gilpin, and David Levine, 'CABOB: A fast optimal algorithm for combinatorial auctions', in *IJCAI*, pp. 1102–1108, (2001).
- [11] D. Schuurmans, F. Southey, and R. C. Holte, 'The Exponential Subgradient Algorithm for Heuristic Boolean Programming', in *IJCAI*, (2001).
- [12] Holger H. Hoos, and Craig Boutilier, 'Solving Combinatorial Auctions Using Stochastic Local Search', in *AAAI/IAAI*, pp. 22-29, (2000).

Nearest neighbor technique and artificial neural networks for short-term electric consumptions forecast

Van Giang Tran, Stéphane Grieu¹ and Monique Polit

Laboratoire ELIAUS, Université de Perpignan Via Domitia, France

Abstract. Promoting both energy savings and renewable energy development are two objectives of the actual and national French energy policy. In this sense, the present work takes part in a global development of various tools allowing managing energy demand. So, this paper is focused on estimating short-term electric consumptions for the city of Perpignan (south of France) by means of the Nearest Neighbor Technique (NNT) or Kohonen self-organizing map and multi-layer perceptron neural networks. The analysis of the results allowed comparing the efficiency of both used tools and methods. Future work will first focus on testing other popular tools for trying to improve the obtained results and secondly on integrating a forecast module based on the present work in a virtual power plant for managing energy sources and promoting renewable energy.

Keywords. Short-term electric consumption, the nearest neighbor technique, Kohonen self-organizing map, multi-layer perceptron, virtual power plant.

1. Introduction

Managing energy demand, extending supply and production sources, developing research activities and ensuring the existence of both transport infrastructure and energy storage are the major challenges of the actual and national French energy policy. Energy demand management entails actions that influence the quantity, or patterns of use, of energy consumed by end users, such as actions targeting reduction of peak demand during periods when energy-supply systems are constrained. Two additional and complementary objectives are, first, promoting energy savings, especially in sectors of daily use (housing, offices, shops and transport) where consumption is high and, secondly, promoting renewable energy (biomass, solar power, geothermal energy and heat pumps) [1].

Mankind's traditional uses of wind, water, and solar energy are widespread in developed and developing countries, but the mass production of electricity using renewable energy sources has become more commonplace recently, reflecting the major threats of [climate change](#), exhaustion of [fossil fuels](#), and the environmental, social and political risks of fossil fuels. Consequently, many countries, like France,

¹Correspondence to: Stéphane Grieu, Laboratoire ELIAUS, Université de Perpignan Via Domitia, 52 Avenue Paul Alduy, 66860 Perpignan, France. Tel.: +33 4 68 66 22 02; Fax: + 33 4 68 66 22 87; E-mail: grieu@univ-perp.fr.

promote renewable energies through tax incentives and subsidies. France has few indigenous sources, only of small amounts of coal, oil and gas. The exploitation of these resources has steadily decreased over the last two decades and nuclear power has dominated the energy supply market [2].

The present work takes part in a global development of reliable and robust tools allowing managing energy sources and promoting renewable energy. In this sense, short-term electric consumptions forecast is of paramount interest and has led to various work during the past years [3,4,5,6]. So, this paper is focused, as a first approach, on estimating short-term electric consumptions for the city of Perpignan by means of well-known and easy to use tools like the Nearest Neighbor Technique (NNT) or Kohonen self-organizing map and multi-layer perceptron neural networks. Using these three tools for estimating short-term electric consumptions over the next few days (the current day and the next two days) is based on the only knowledge of both previous days consumptions (two to four days; taking into account more than the four previous days consumptions has been tested but proved to be unnecessary because, whatever the used tool, it did not improve the obtained forecasts) and previous year consumptions (historical data). The analysis of the results allowed comparing the efficiency of the used tools and methods.

No meteorological parameters have been used but future work will focus, first, on taking into account this kind of parameters and, secondly, on considering other and perhaps more effective methods like, for example, the Box-Jenkins (allowing to obtain ARIMA models) [7,8] and the FIR (Fuzzy Inductive Reasoning) [9] methods. Finally, a forecast module based on the present and on future but complementary work will be integrated in a virtual power plant for managing energy sources and promoting renewable energy.

2. Study area : the Perpignan city

The chosen study area is the Perpignan city. It is the administrative capital city of the Pyrénées-Orientales department in southern France. Perpignan is around 150 km from Montpellier, 200 km from Toulouse or Barcelona (Spain) and 850 km from Paris. Its population is about 120 000 in the city proper and more than 300 000 in the metropolitan area. Because of its proximity with the Mediterranean Sea, its population highly increases during the summer season.

A Mediterranean climate is found in Perpignan: high temperature in summer and mild climate for the rest of the year. The weather is globally hot, with dry summers, mild and humid winters and just a few rainy days during the year. Perpignan is a very sunny and windy city and is a good place to develop the electricity production by means of photovoltaic solar cells and/or windmills.

3. Materials and methods

3.1. Daily electric consumptions database

Perpignan's daily electric consumptions for years 2005 and 2006 allowed carrying out short-term forecasts by means of the nearest neighbor technique and both Kohonen self-organizing map and multi-layer perceptron neural networks. The database is

composed of 730 daily electric consumptions (104 weeks over the two above-mentioned years), expressed in MW per day. A short and preliminary data analysis revealed the presence of cyclic dynamics in terms of trends and seasonality.

In the present work, 2005 data have been used as historical data and 2006 data for testing various tools in the field of daily electric consumptions forecasts. Data were provided by RTE (Gestionnaire du Réseau de Transport d'Electricité), an EDF (Electricité De France) department with independent accounts, managements and finances. RTE's main objectives are about managing network infrastructures, managing electricity flows on the network and contributing to the proper running of the electricity market [6].

3.2. The nearest neighbor technique

In [statistics](#), [signal processing](#), and [econometrics](#), a time series is a sequence of [data points](#), measured typically at successive times, spaced at (often uniform) time intervals. Time series analysis comprises methods that attempt to understand such time series [10], often either to understand the underlying theory of the data points, or to make [forecasts](#) (predictions). So, time series analysis allows predicting future events based on known past events: to predict future data points before they are measured.

In the present work, a classical and well-known tool has been first used: the nearest neighbor technique, based on time series proximity and similarity [11].

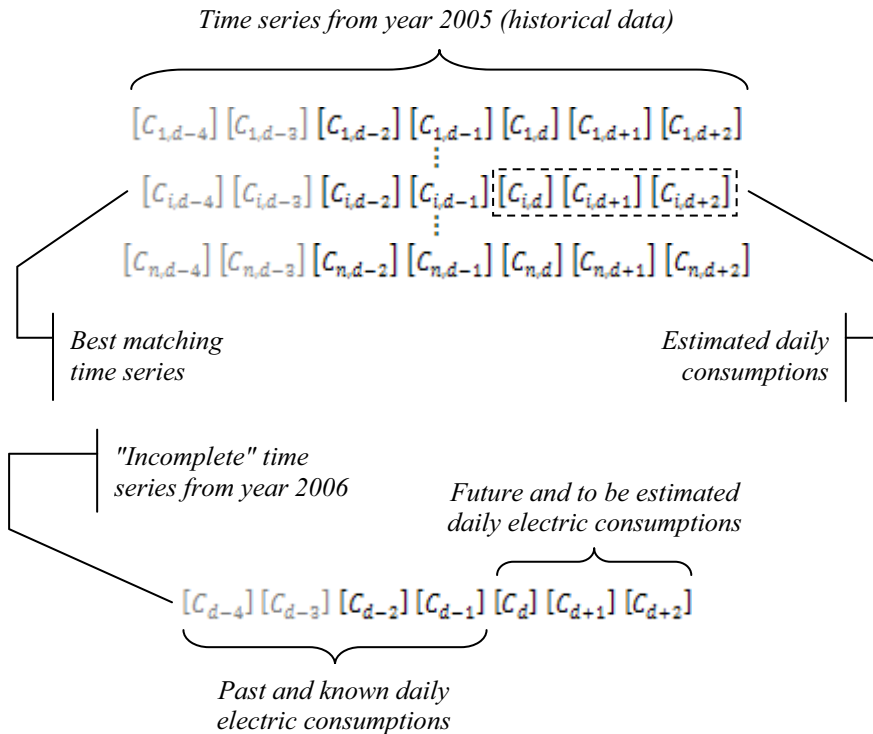


Figure 1. The nearest neighbor technique for short-term electric consumptions forecast.

Historical data are used as the basis of estimating future outcomes. The most widely used measures of time series proximity are the Euclidian distance and dynamic time warping. For this work, the daily electric consumption database has been cut in time series whose sizes depend, first, on the chosen forecast horizon (here the current and the next two days electric consumptions $[C_d]$, $[C_{d+1}]$ and $[C_{d+2}]$), secondly, on the concerned days of the week (Monday and/or Tuesday and/or ... and/or Sunday) and, thirdly, on the previous daily electric consumptions taken into account (only the two previous days electric consumptions $[C_{d-1}]$ and $[C_{d-2}]$ or the three previous days electric consumptions $[C_{d-1}]$, $[C_{d-2}]$ and $[C_{d-3}]$ or the four previous days electric consumptions $[C_{d-1}]$, $[C_{d-2}]$, $[C_{d-3}]$ and $[C_{d-4}]$). As previously mentioned, taking into account more than the four previous days electric consumptions proved to be unnecessary and did not improve the obtained forecast results. Following these criteria, a time series can be composed of 5 to 7 daily electric consumptions (the optimal size for time series is given by the best obtained forecasts). By calculating the Euclidian distance between an "incomplete" time series from year 2006 (i.e. including future and to be estimated daily electric consumptions) and each of the time series from year 2005, the best matching time series can be highlighted and provides the estimated daily electric consumptions (Figure 1).

3.3. The Kohonen self-organizing map (KSOM)

3.3.1. Network structure

The Kohonen self-organizing map consists of a regular, usually two-dimensional, grid of neurons (output neurons). Each neuron i is represented by a weight, or model vector, $m_i = [m_{i1}, ..., m_{in}]^T$ where n is equal to the dimension of the input vectors. The set of weight vectors is called a codebook.

The neurons of the map are connected to adjacent neurons by a neighborhood relation, which dictates the topology of the map. Usually rectangular or hexagonal topology is used. Immediate neighbors belong to the neighborhood N_i of the neuron i . The topological relations and the number of neurons are fixed before the training phase allowing the map configuration. The number of neurons may vary from a few dozens up to several thousands. It determines the granularity of the mapping, which affects the accuracy and generalization capability of the KSOM [12,13].

3.3.2. Training algorithm

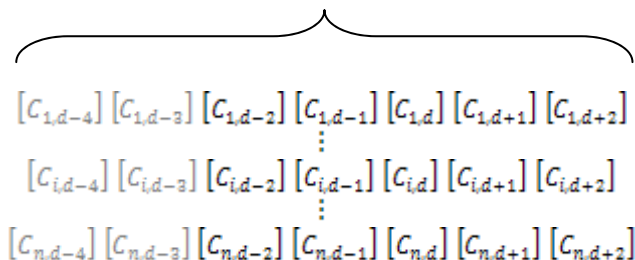
The Kohonen feature map creates a topological mapping by adjusting, for input vectors, not only the winner's weights, but also adjusting the weights of the adjacent output units in close proximity or in the neighborhood of the winner (the Best Matching Unit). So not only does the winner get adjusted, but the whole neighborhood of output units gets moved closer to the input pattern. Starting from randomized weight values, the output units slowly align themselves such that when an input pattern is presented, a neighborhood of units responds to the input pattern. As training progresses, the size of the neighborhood radiating out from the winning unit is decreased. Initially large numbers of output units will be updated, and later on smaller and smaller numbers are updated until at the end of training only the winning unit is adjusted. Similarly, the learning rate will decrease as training progresses, and in some implementations, the learn rate decays with the distance from the winning output unit.

3.3.3. Short-term electric consumption forecast using KSOM

Short-term electric consumption forecast can be done using Kohonen self-organizing maps in the same way that this kind of neural networks is used for estimating missing (unknown) values in a database. The networks, after a training phase carried out using available complete input vectors, can be used for estimating missing values in new input vectors, by means of BMU (Best Matching Unit) weight vectors [12,13].

KSOM Training phase

*Daily electric consumptions vectors from year 2005(historical data)
used for the KSOM training phase*



Forecast phase

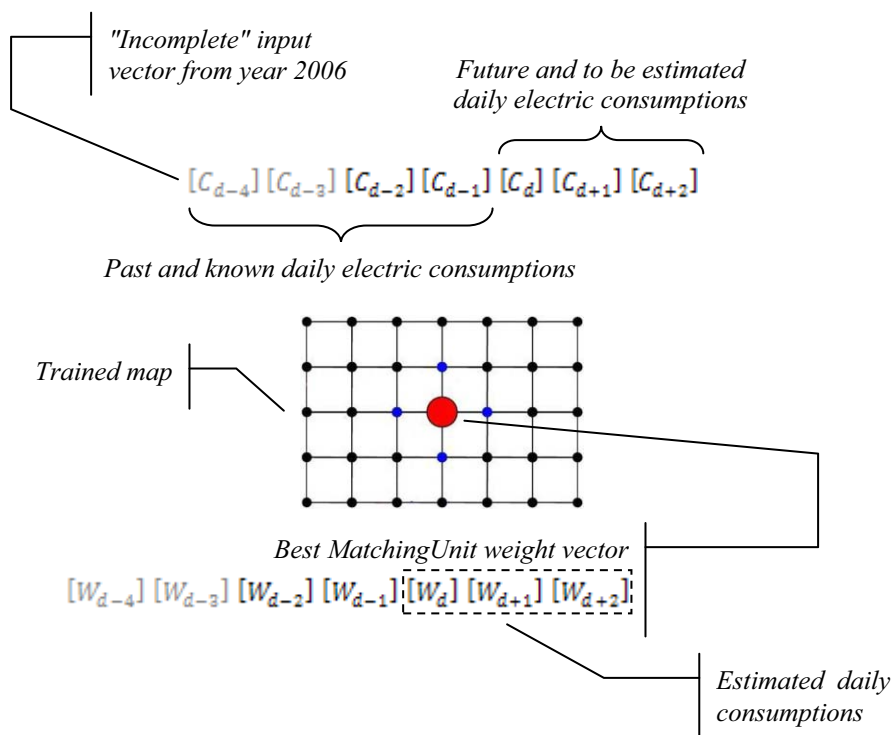


Figure 2. Kohonen self-organizing maps for short-term electric consumptions forecast.

As when using the nearest neighbor technique, the daily electric consumptions database has been cut in vectors composed of 5 to 7 daily electric consumptions (the optimal size for vectors is given by the best obtained forecasts). Vectors from year 2005 have been used for the networks training phase. Forecasts are carried out by means of "incomplete" vectors from year 2006 (i.e. including future and to be estimated daily electric consumptions) used as networks new input vectors. For each "incomplete" input vector, the BMU is determined and its weight vector provides the estimated daily electric consumptions (Figure 2).

3.4. The multi-layer perceptron (MLP)

3.4.1. Network structure

The perceptron, the simplest neural network, is only able to classify data into two classes. Basically it consists of a single neuron with a number of adjustable weights. It uses an adaptative learning rule. Given a problem which calls for more than two classes, several perceptrons can be combined: the simplest form of a layered network just has an input layer of source nodes that connect to an output layer of neurons.

The single-layer perceptron can only classify linearly separable problems. For non-separable problems it is necessary to use more layers. A multi-layer network has one or more hidden layers whose neurons are called hidden neurons. The network is fully connected; every node in a layer is connected to all nodes in the next layer.

According to the previous remarks, the network used for the present work is a multi-layer perceptron. It consists of one layer of linear output neurons and one hidden layer of nonlinear neurons [14]. According to previous tests, more than one hidden layer proved to cause slower convergence during the learning phase because intermediate neurons not directly connected to output neurons learn very slowly. Based on the principle of generalization versus convergence, the number of hidden neurons and the number of iterations completed during the training phase were optimized.

3.4.2. The backpropagation algorithm

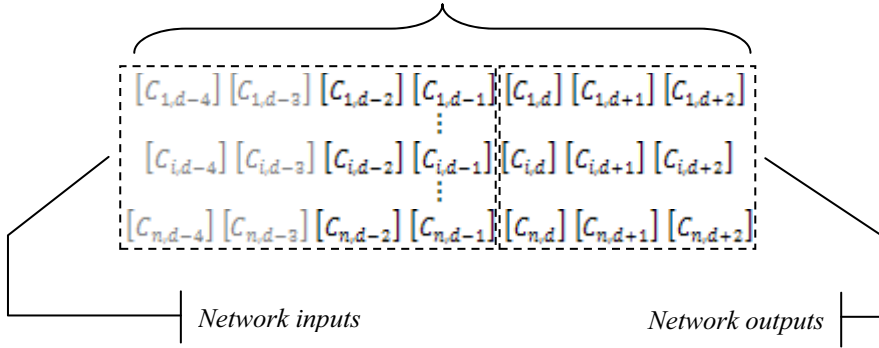
The multi-layer perceptron neural network learns using an algorithm called backpropagation. During this iterative process, input data are repeatedly presented to the network. With each presentation, the network output is compared to the desired output and an error is computed. This error is then fed back to the network and used to adjust the weights such that it decreases with each iteration and the model gets closer and closer to produce the desired output [15].

3.4.3. Short-term electric consumption forecast using MLP

As when using the nearest neighbor technique, the daily electric consumptions database has been cut in vectors composed of 5 to 7 daily electric consumptions (the optimal size for vectors is given by the best obtained forecasts). Vectors from year 2005 have been used for the networks training phase: depending on vectors sizes, two to four daily electric consumptions were used as network inputs and three as network outputs. Forecasts are carried out by means of "incomplete" vectors from year 2006 (i.e. including future and to be estimated daily electric consumptions) used as networks new input vectors (Figure 3).

MLP Training phase

*Daily electric consumptions vectors from year 2005 (historical data)
used for the MLP training phase*



Forecast phase

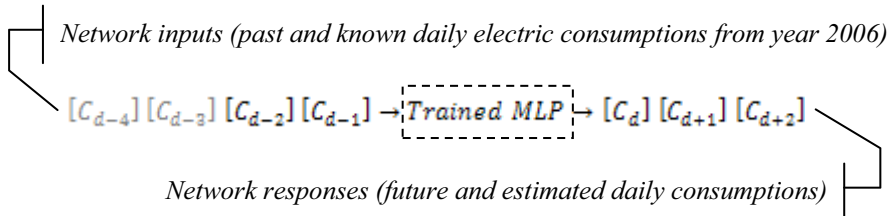


Figure 3. Multi-layer perceptron for short-term electric consumptions forecast.

4. Forecast results

Tables 1, 2 and 3 only present some of the obtained results (Mon. 23 to Wed. 25 Jan. 2006, Sat. 23 to Mon. 25 Sep. 2006 and Sun. 10 to Tue. 12 Dec. 2006) for short-term electric consumptions forecast using the nearest neighbor technique (Table 1) and both Kohonen self-organizing map (Table 2) and multi-layer perceptron (Table 3) neural networks. The presented results are representative samples of all the obtained results covering year 2006. According to time series and vectors sizes (i.e. according to the previous daily electric consumptions taken into account), for all the estimated electric consumptions, relative errors have been calculated using all the corresponding real values. Overall mean relative errors have also been calculated.

The first conclusion of the present work is that all the used tools and methodologies are useful in the field of short-term (here the current day and the next two days) electric consumptions forecast. They have been successfully applied to the city of Perpignan. However, both used neural networks (KSOM and MLP) provided better results (i.e. lower overall mean relative errors) than the nearest neighbor technique: 2 to 3 % vs. 7 to 9 %.

Another conclusion is that using the nearest neighbor technique, one can notice that, whatever the time series sizes (7, 6 or 5), relative errors increase according to the forecast horizon (the current day, the next day or the next two days). However, nothing

similar can be noticed and highlighted using KSOM and MLP neural networks: whatever the above-mentioned forecast horizon, mean relative errors are low.

Table 1. Short-term electric consumptions forecast using the nearest neighbor technique.

| Forecast dates | Time series sizes: 7 | | Time series sizes: 6 | | Time series sizes: 5 | |
|---------------------------------|----------------------|-------------------------|----------------------|-------------------------|----------------------|-------------------------|
| | Relative error (%) | Mean relative error (%) | Relative error (%) | Mean relative error (%) | Relative error (%) | Mean relative error (%) |
| Mon. 01/23/06 | 5.96 | | 5.96 | | 5.96 | |
| Tue. 01/24/06 | 10.81 | 9.46 | 10.81 | 9.46 | 10.81 | 9.46 |
| Wed. 01/25/06 | 11.63 | | 11.63 | | 11.63 | |
| Sat. 09/23/06 | 4.55 | | 7.82 | | 7.82 | |
| Sun. 09/24/06 | 4.7 | 5.14 | 9.36 | 9.91 | 9.36 | 9.91 |
| Mon. 09/25/06 | 6.18 | | 12.55 | | 12.55 | |
| Sun. 12/10/06 | 4.46 | | 4.46 | | 0.74 | |
| Mon. 12/11/06 | 5.69 | 6.49 | 5.69 | 6.49 | 9.29 | 7.66 |
| Tue. 12/12/06 | 9.33 | | 9.33 | | 12.95 | |
| Overall mean relative error (%) | 7.03 | | 8.62 | | 9.01 | |

Table 2. Short-term electric consumptions forecast using KSOM.

| Forecast dates | Vectors sizes: 7 | | Vectors sizes: 6 | | Vectors sizes: 5 | |
|---------------------------------|--------------------|-------------------------|--------------------|-------------------------|--------------------|-------------------------|
| | Relative error (%) | Mean relative error (%) | Relative error (%) | Mean relative error (%) | Relative error (%) | Mean relative error (%) |
| Mon. 01/23/06 | 1.31 | | 1.31 | | 0.87 | |
| Tue. 01/24/06 | 4.85 | 2.55 | 3.32 | 1.72 | 1.16 | 2.48 |
| Wed. 01/25/06 | 1.5 | | 0.53 | | 5.41 | |
| Sat. 09/23/06 | 3.55 | | 1.61 | | 3.28 | |
| Sun. 09/24/06 | 3.1 | 3.06 | 2.76 | 2.61 | 3.11 | 3.14 |
| Mon. 09/25/06 | 2.53 | | 3.47 | | 3.03 | |
| Sun. 12/10/06 | 4.17 | | 0.09 | | 0.82 | |
| Mon. 12/11/06 | 0.72 | 2.71 | 1.86 | 2.51 | 4.65 | 3.95 |
| Tue. 12/12/06 | 3.23 | | 5.6 | | 6.37 | |
| Overall mean relative error (%) | 2.77 | | 2.28 | | 3.19 | |

Table 3. Short-term electric consumptions forecast using MLP.

| Forecast dates | Vectors sizes: 7 | | Vectors sizes: 6 | | Vectors sizes: 5 | |
|---------------------------------|--------------------|-------------------------|--------------------|-------------------------|--------------------|-------------------------|
| | Relative error (%) | Mean relative error (%) | Relative error (%) | Mean relative error (%) | Relative error (%) | Mean relative error (%) |
| Mon. 01/23/06 | 2.35 | | 1.6 | | 5.25 | |
| Tue. 01/24/06 | 2.56 | 2.30 | 2.08 | 1.24 | 1.35 | 2.73 |
| Wed. 01/25/06 | 1.98 | | 0.05 | | 1.58 | |
| Sat. 09/23/06 | 1.84 | | 1.61 | | 0.9 | |
| Sun. 09/24/06 | 1.6 | 1.45 | 1.12 | 0.94 | 0.15 | 0.36 |
| Mon. 09/25/06 | 0.9 | | 0.09 | | 0.01 | |
| Sun. 12/10/06 | 0.02 | | 6.03 | | 2.14 | |
| Mon. 12/11/06 | 1.54 | 2.68 | 1.68 | 4.24 | 3.25 | 3.64 |
| Tue. 12/12/06 | 6.47 | | 5.02 | | 5.55 | |
| Overall mean relative error (%) | 2.14 | | 2.14 | | 2.24 | |

The next conclusion is that using the four previous days electric consumptions, instead of the two or the three previous, for carrying out the forecasts does not significantly improve the obtained results, except when using the nearest neighbor technique.

5. Conclusions

The present work takes part in a global development of reliable and robust tools allowing managing energy sources and promoting renewable energy. In this sense, short-term electric consumptions forecast is of paramount interest. So, this paper is focused on estimating short-term electric consumptions (the current day and the next two days) for the city of Perpignan by means of the nearest neighbor technique and both Kohonen self-organizing map and multi-layer perceptron artificial neural networks. Only previous days and previous year daily electric consumptions allowed carrying out the forecasts.

Conclusions of the present work are that all the used tools and methodologies are useful in the field of short-term electric consumptions forecast but both used neural networks (KSOM and MLP) provided better results than the nearest neighbor technique. Moreover, taken into account more than the two previous days electric consumptions does not significantly improve the obtained results, except for the nearest neighbor technique.

Current work focus now on determining the maximal forecast horizon (it seems that the results start to deteriorate for a forecast horizon set to three days from the current day) and future work will focus, first, on taking into account meteorological parameters and, secondly, on considering other and perhaps more effective methods like, for example, the Box-Jenkins and/or the FIR (Fuzzy Inductive Reasoning) methods. However, improving the obtained results using other and more sophisticated methods seems to be difficult. Finally, a forecast module based on the present and on future but complementary work will be integrated in a virtual power plant for managing and promoting renewable energy.

References

- [1] ADEME - French state agency for the environment and energy conservation, <http://www.ademe.fr>.
- [2] RTE - Transmission System Operator in France, http://www.rte-france.com/index_en.jsp.
- [3] Gross G. and Galiana F. D., Short-term load forecasting, *Proc. IEEE* 75 (12), pp. 1558-1573, 1987.
- [4] Kandil N., Wamkeue R., Saad M. and Georges S., An efficient approach for short term load forecasting using artificial neural networks, *Int. Journal of Electrical Power & Energy Systems* 28 (8), pp. 525-530, 2006.
- [5] Park D. C., El-Sharakawi M. A. and Marks II Ri., Electric load forecasting using artificial neural networks, *IEEE Trans. Power Sys.* 6 (2), pp.442-449, 1991.
- [6] Dillon T. S., Morsztyn K. and Phua K., Short term load forecasting using adaptative pattern recognition and self-organizing techniques, in: *Proceedings of the fifth world power system computation conference (PSCC-5)*, Cambridge, paper 2.4/3, pp. 1-15, 1975.
- [7] Box G.E.P. and Jenkins G.M., *Time series analysis: Forecasting and control*, Holden-Day, San Francisco, 1970.
- [8] Makridakis S., Wheelwright S.C. and Hyndman R.J., *Forecasting: methods and applications*, John Wiley & sons, New York, 1998.
- [9] Cellier F. E., Nebot A., Mugica F. and De Albornoz A., Combined qualitative/quantitative simulation models of continuous-time processes using fuzzy inductive reasoning techniques, *Int. Journal of General Systems* 24 (1-2), pp. 95-116, 1996.
- [10] Brockwell P. J. and Davis R. A., *Introduction to Time Series and Forecasting*, Springer, 1997.

- [11] Brockwell P. J. and Davis R. A., *Time Series: Theory and Methods*, 2nd edition, Springer 1991.
- [12] Simula O., Alhoniemi E., Hollmén J. and Vesanto J., Analysis of complex systems using the self-organizing map, in: *Proceedings of 4th International Conference on Neural Information Processing (ICONIP'97)*, Dunedin, New Zealand, pp. 1313-1317, 1997.
- [13] Simula O., Vesanto J., Alhoniemi E. and Hollmén J., Analysis and modeling of complex systems using the self-organizing map, in: N. Kasabov, R. Kozma (Eds.), *Neuro-Fuzzy Techniques for Intelligent Information Systems*, Pysica-Verlag, pp. 3-22, 1999.
- [14] Hornik K., Stinchcombe M. and White H., Multi-layer feedforward networks are universal approximation, *Neural Networks* 2, pp. 359-366, 1989.
- [15] Rumelhart D.E., Hinton G.E. and Williams R.J., Learning internal representations by error propagation, in: Rumelhart D.E. and McClelland J.L. (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition*, Vol.1, MIT Press, Cambridge, pp. 318-364, 1986.

Pattern Discovery in Melanoma Domain Using Partitional Clustering

David VERNET^a, Ruben NICOLAS^a, Elisabet GOLOBARDES^a,
Albert FORNELLS^a, Carles GARRIGA^a, Susana PUIG^b and Josep MALVEHY^b

^a *Grup de Recerca en Sistemes Intel·ligents
Enginyeria i Arquitectura La Salle, Universitat Ramon Llull
Quatre Camins 2, 08022 Barcelona (Spain)
{dave,rnicolas,elisabet,afornells,cgarriga}@salle.url.edu*

^b *Melanoma Unit, Dermatology Department
IDIBAPS, U726 CIBERER, ISCIII
Hospital Clinic i Provincial de Barcelona (Spain)
{spuig,jmalvey}@clinic.ub.es*

Abstract. Nowadays melanoma is one of the most important cancers to study due to its social impact. This dermatologic cancer has increased its frequency and mortality during last years. In particular, mortality is around twenty percent in non early detected ones. For this reason, the aim of medical researchers is to improve the early diagnosis through a best melanoma characterization using pattern matching. This article presents a new way to create real melanoma patterns in order to improve the future treatment of the patients. The approach is a pattern discovery system based on the K-Means clustering method and validated by means of a Case-Based Classifier System.

Keywords. Melanomas, Pattern Discovery, Clustering, Artificial Intelligence in Medicine, Computer Aided Systems, Case-Based Reasoning.

Introduction

New social habits in solar exposure make much more important the melanoma early diagnosis due to its high ratio of people affected by this illness. Analyzing data from the American Cancer Society and from specialists in skin cancer we could observe that this type of disease is increasing its rates of appearance. In addition, this kind of skin cancer is which accumulate highest percentage of death in front of more common dermatological cancers, approximately a twenty percent of non early prognosticated cases [9]. Recently diagnosis of melanoma is based on the ABCD rule [7] which considers the following features commonly observed in this kind of tumour: (1) a diameter larger than 5 mm (2) colour variegation (3) asymmetry and (4) border irregularity. Although most of melanomas are correctly diagnosed following this rule, a variable proportion of cases does not comply with these criteria. Therefore, medical researchers focus on identifying reliable melanoma patterns to improve diagnosis because the actual patterns do not offer a good one.

The goal of this paper is to identify a set of reliable patterns from a dataset built by a group of melanomas seniors researchers. The extraction and validation process is performed by ULIC (Unsupervised Learning in CBR) system. The platform explores data from a data intensive approach based on partitional clustering technique.

Clustering algorithms can be classified in two basic types: hierarchical and partitional clustering. Hierarchical clustering proceeds successively by either merging smaller clusters into larger ones, or by splitting large clusters. The different algorithms based on this type of clustering differ in the way which two small clusters are merged or which large cluster is split. On the other hand, partitional clustering attempts to directly decompose the data set into a set of disjoint clusters. The choice of clustering algorithm depends on the type of data available and on the particular purpose and application [10]. In our case, we want to identify disjointed regions in the domain associated to a certain pattern. For this reason partitional algorithms are the most suitable for this purpose.

On the hand, the validation process is assessed through a Case-Based Reasoning (CBR) [1] approach. The main benefit of this approach is its capability for explaining why the classification has been done. Therefore, this issue is crucial helping experts to understand the results.

This paper is organized as follows. In section 1 the related work is presented. Section 2 describes the framework. Section 3 summarizes the experimentation and results. Section 4 exposes the conclusions and further work.

1. Related Work

Clustering techniques are a smart way to extract relationships from huge data amounts. Consequently, this useful property has been widely used in medical domains [6]. The types of work mainly depend on the data topology and the usage of extracted relations from analysis. As we have commented in previous section the Partitional Clustering is the most suitable approach for our purpose.

Two points of view can be taken in the Partitional clustering. The first one is the hard clustering, where each case is assigned to only one cluster. Second one, is the fuzzy clustering where each case is assigned a degree of membership of between 0 and 1 to each cluster. One of the main partitional hard clustering technique is the K -means algorithm [13]. There are special variations to improve some aspects of the algorithm. One of these is the K -medoids algorithm or PAM (Partition Around Medoids) [12].

There are melanoma studies focused in the identification of relationships between malignant melanoma and familiar or hereditary tumours (i.e. breast cancer, ovarian cancer, colon cancer, pancreatic cancer) such as in [15]. On the other hand, others works analyse thousands of genes with the aim of extracting the 'guilty' genes [5,4] related to the cancer. Anyway, both approaches help experts to be aware and detect melanoma formation in early stages.

Some works, like [11], have studied techniques available for the validation of clustering results. In particular, in [2] the standard measure of interset distance (the minimum distance between points in a pair of sets) was computed. It was the least reliable measure and experimental results also suggest that intercluster separation plays a more important role in cluster validation than cluster diameter. Dunn's indexes provided good validation results.

We remark the importance of CBR in Computer Aided Systems for medical investigation, as has been studied in psychiatry [3], renal failure cases [14], breast cancer [8] and so on. But nowadays it seems interesting to apply it to melanomas cancer attending to its increasing importance.

2. Discovering melanoma patterns with ULIC platform

One of the applications of ULIC (Unsupervised Learning In Case-Based Classifier Systems) platform (author cited) is to discover patterns and evaluate them. It contains a set of unsupervised methods which allow to explore non labelled data in order to create groups of similar examples represented by a prototype. Each prototype is what system considers as one possible class. In this paper, the aim is to extract patterns with usefulness for medical researchers based on the clusterization of melanomas information. Next, a second phase using CBR is applied in order to validate these patterns.

Medical researchers want to know what is the optimal number of melanoma patterns and characterize each one of them. For this reason, we have tackled the patterns discovery process with the K -means algorithm [13] because it allows us to perform an incremental search using different number of clusters, even other approach could be tested. Next, with the aim of checking the quantitative goodness of these new classifications, we use CBR system to assess the performance of the patterns discovered.

Next sections introduce CBR, k -means algorithm, and how they are integrated.

2.1. Case-Based Reasoning

CBR is a technique that solves new cases using others previously solved. In order to get this objective four phases are applied [1]. 1) Retrieve the most similar cases from the case memory with the assistance of a similarity function. 2) Try to reuse the solutions from the retrieved cases with the aim to solve the present case. 3) Make a revision of the solution. 4) Retain the useful information of the solved case, if it is necessary.

All steps turns around the Case Memory, which contains the experience of system in terms of cases. A case is a description of a problem.

2.2. K -means clustering

K -means is an unsupervised clustering method [13] that permits to do a partition of the domain in K clusters. The algorithm can be described as follows:

- 1: Choose an initial partition of the cases into k clusters. This is random assignment to k clusters.
- 2: Compute the distance from every case to the mean of each cluster and assign the cases to their nearest clusters.
- 3: Recompute the cluster means following any change of cluster membership at step 2.
- 4: Repeat steps 2 and 3 until no further changes of cluster membership occur in a complete iteration. The procedure has now converged to a stable k -partition.

At the end of the process, the mean value of the cases assigned to the cluster defines the centroid of the cluster, which represents the general properties of cases.

The distance function used in *k*-means is based on the Euclidean distance. For the discrete attributes, the most repeated value is used as mean value and the Heterogenous Value Difference Metric (HVDM) [16].

In order to use the Euclidean distance we get the assumption that the attributes are statistically independent and they vary in an equal way in clusters with a hiperspheric shape. More over, all clusters contain at least one case, so no empty clusters are obtained after applying the algorithm.

2.3. Integration of *K*-means and CBR in ULIC

ULIC is a system capable of discovering patterns in a specific domain. The platform integrates a system for exploring the different groupings with different number of clusters (*K*-means) and another one for evaluating the possible solutions we have obtained (CBR) (see Fig. 1).

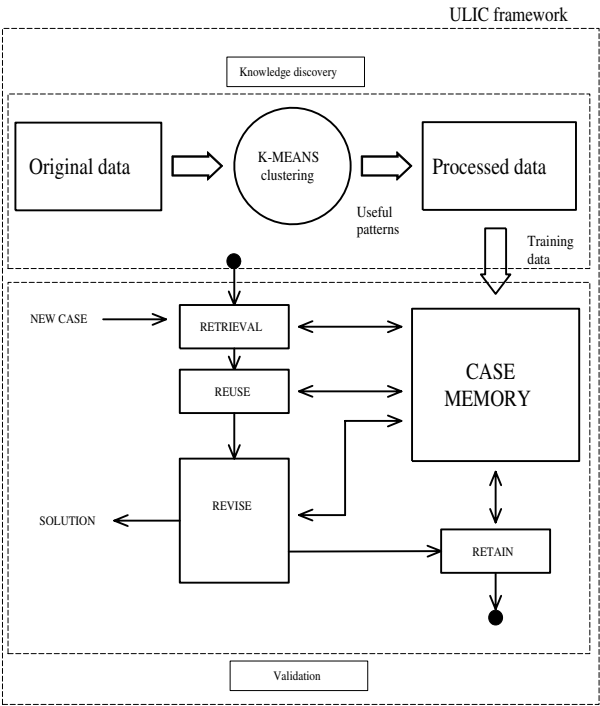


Figure 1. ULIC framework.

Let us have a look to each system module. First of all, we find the module where the data is processed. At this point, K-means clustering is applied to input data in order to discover new patterns in data structure. Using the clusters obtained by this module, the training cases are redefined and their classes are changed to the new ones.

The second module consists of a traditional CBR classifier that uses the 4-R cycle. According to the cluster in which they are the new data organization is introduced in the

Case Memory and the new training set is generated. CBR is used so as to validate the new clusters obtained in the case memory.

It is important to notice that it is not necessary to apply the CBR cycle when the medical staff need information about the patterns. Using the same technique, they could extract this information from the Memory Case using the previously classified cases.

An aspect to comment is that the labels of each cluster are unknown and it is not possible to generate a training data set and a test data set with the original data without avoiding the loss of information. Clustering is applied to whole data.

3. Experiments: Results and Discussion

As we have explained in previous sections, the aim of this work is to discover new patterns related to the melanoma domain to help medical staff in the early diagnosis. The next sections describe the dataset defined by medical experts and the melanoma patterns defined.

3.1. Data Description

The medical data used in our experimentation is extracted using real cases from the Melanoma Unit in *Hospital Clinic i Provincial de Barcelona*. The kind of data we have used is from two types: Dermatological, which represents the information gathered by dermatologists (that represents very specific knowledge) and Histological, which is the data collected by nurses (include more generic information). On the other hand, we could see that dataset is very varied and includes numerical and categorical values. We would like to remark that the attributes with categorical values, in spite of being integer values, can not be ordered for its value because it works as a label. The concrete domain and data type for each attribute are shown at Table 1. Although dataset is small (70 instances), it is considered by experts as an enough representation.

The next section focus on defining new classification for each sample.

Table 1. Attributes Domain and Types.

| Attribute | Domain | Type |
|-----------------------|--|----------------|
| sex | Male, Female | histological |
| age | numeric | histological |
| max diam | numeric | histological |
| site | trunk,lower Extr,upper Extr,back,leg, arm,forearm,sovraclav,neck,shoulder | histological |
| pigment network | 0,1,2 | dermatological |
| dots and globules | 0,1,2 | dermatological |
| streaks | 0,1,2 | dermatological |
| regression structures | 0,1,2,3,4 | dermatological |
| BW veil | 0,1 | dermatological |
| bloches | 0,1,2 | dermatological |
| vessels | 0,1,2,3,4 | dermatological |
| millia like cyst | 0,1 | dermatological |

3.2. Discovering New Patterns

With the aim of improving the possibility of getting a right diagnosis, we have used an incrementally exploration of K in the K-means algorithm. This incremental execution involves the analysis of classification between two and twenty different classes (clusters) and ten seeds for each one. The experts considered that 20 classes were sufficient for a so low number of instances.

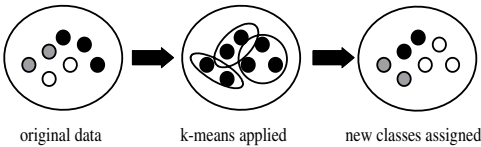


Figure 2. New dataset creation process. Cases are relabelled according to the cluster in which they belong

After having these results, for each seed from each number of classes, we have created a new dataset with the same instances as the initial one but with a new class attribute, attending to the cluster it has been assigned in (Fig. 2) previously. By the use of this new set, we have studied the percentage of the error rate of the CBR classification. The validation of the new classes is based on the execution of a CBR cycle setting Nearest Neighbor as retrieval phase and leave-one-out (due to the lack of original cases).

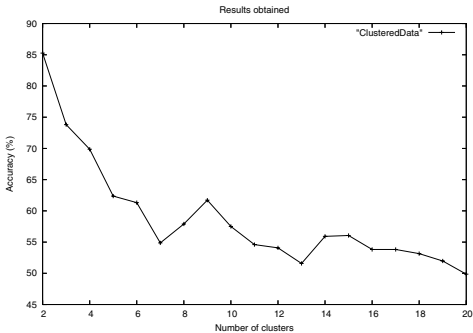


Figure 3. Results obtained using the artificial clusters. Ten different seeds are used for each number of clusters. Mean accuracy is represented for each k .

Fig. 3 summarizes the results as the average accuracy rate of the different seeds of every classification. The analysis of the results shows that the classification pattern, obtained with the K-means part of ULIC, permits an accuracy of at least fifty percent. The superior limit of correctly classification is around eighty-five percent. Both superior and inferior limits correspond to the smaller number of cases (2 clusters) and the biggest one (20 clusters), respectively.

The decreasing tendency has an exception for nine clusters, when an increase of approximately ten percent of accuracy is obtained.

In summary, we could say that focusing the attention on a quantitative analysis, we could remark four important ideas: 1) The classification results are better with not

many clusters. 2) We have a local maximum with nine clusters. 3) The results remark a decreasing tendency. 4) A recommendation of new classification criteria is to use nine classes.

Analyzing the patterns obtained by a 9-Means classification we can observe the following information (just informative):

- Cluster 0: The attributes *bloches* and *millia_like_cyst* are always 0. The site is always different of *shoulder*, *neck*, *arm* or *upper – extr.*
- Cluster 1: The attribute *Max_diam* is never large or huge. The attributes *pigment_network*, *streaks*, *BW_veil* and *millia_like_cyst* are always 0. *Dots_and_globules* is always 2.
- Cluster 3: *Vessels* is always different of 2, 3 and 4. The *site* is different of *sovraclav* and *shoulder*.
- Cluster 4: The *sex* is always male. None are young. *BW_veil*, *bloches*, *vessels* and *millia_like_cyst* are always 0. *Pigment_network* is always 2.
- Cluster 5: *BW_veil* and *millia_like_cist* are always 0. Attributes *Bloches* and *Pigment_network* are always different of 0 and 2 respectively.
- Cluster 6: *Regression_structures* and *BW_veil* are always 0. *Streaks* is always different of 1. *Dots_and_globules* is always different of 2.
- Cluster 7: All are male. *Dots_and_globules*, *BW_veil*, *bloches*, *vessels* and *millia_licke_cyst* are always 0. None are huge or large (*Max_diam*).
- Cluster 8: All are male. *Pigment_network* is always 2. *BW_veil*, *vessels*, *millia_like_cyst* are always 0. *Max_diam* is always different of small and huge.

Attending to the evaluation performed by the medical staff, which is a 8-classes classification, we propose to analyze now the possibility of classifying with nine classes instead of the eight used until now.

A preliminary study performed by medical staff of these clusters showed interesting groups of data. This study has to be complemented subsequently by medical researchers in a medical way, adding meaning to the patterns we have extracted.

4. Conclusions and Further Work

Due to new habits in solar exposure, melanoma is increasing its appearance. Early diagnosis is crucial in order to assault the recovery with successful possibilities. In this way it is important to tackle the search of accurate melanoma patterns. Achieving these patterns could permit non expert doctors to easy recognize suspicious injuries. This work uses ULIC to extract these patterns analyzing real medical data. ULIC is a platform for discovering data patterns through clustering techniques. Moreover, ULIC is capable to validate the extracted patterns using a CBR system. The main benefit of CBR is its capability for explaining the results to the experts.

The results of the experimentation provide experts an encouraging point of view in the discovering of melanoma patterns due to the promising results achieved.

The further work is focus on analyzing the medical utility of the new patterns. Medical staff should now analyze the possibility of classifying with nine classes instead of the eight used till now. At this point, we should corroborate the patterns with the comments of senior medical researchers.

Acknowledgements

We would like to thank the Spanish Government for the support in MID-CBR project under grant TIN2006-15140-C03 and the Generalitat de Catalunya for the support under grants 2005SGR-302 and 2007FI-A 01328. Also, we would like to thank Enginyeria i Arquitectura La Salle of Ramon Llull University for the support to our research group. We also would like to thank to the clinicians involved in the confection of the dataset: Dr Paolo Carli (dermatologist), Dr Vinzenzo di Giorgi (dermatologist), Dr Daniela Massi (dermatopathologist) from the University of Firenze; Dr Josep Malvehy (dermatologist), Dr Susana Puig (dermatologist) and Dr Josep Palou (dermatopathologist) from Melanoma Unit in Hospital Clinic i Provincial de Barcelona. Part of the work performed by S. Puig and J. Malvehy is partially supported by: Fondo de Investigaciones Sanitarias (FIS), grant 0019/03 and 06/0265; Network of Excellence, 018702 GenoMel from the CE.

References

- [1] A. Aamodt and E. Plaza. Case-Based Reasoning: Foundations Issues, Methodological Variations, and System Approaches. In *AI Communications*, volume 7, pages 39–59, 1994.
- [2] J. C. Bezdek and N. R. Pal. Some new indexes of cluster validity. 28(3):301–315, June 1998.
- [3] I. Bichindaritz. A case-based assistant for clinical-psychiatry expertise. *Journal Of The American Medical Informatics Association*, pages 673–677, 1994.
- [4] H. Dang, T. Segaran T. Le, and J. Levy. Integrating database information in microarray expression analyses: Application to melanoma cell lines profiled in the nci60 data set. *Journal of Biomolecular Techniques*, 13:199–204, 2002.
- [5] M.B. Eisen, P.T. Spellman, P.O. Brown, and D. Botstein. Cluster analysis and display of genome-wide expression patterns. *National Academy Scientific*, 25(95):14863–14868, 1998.
- [6] A. Fornells, E. Armengol, E. Golobardes, S. Puig, and J. Malvehy. Experiences using clustering and generalizations for knowledge discovery in melanomas domain. In *7th Industrial Conference on Data Mining*, volume 5077 of *LNCS*, pages 57–71. Springer-Verlag, 2008.
- [7] R. J. Friedman, D. S. Rigel, and A. W. Kopf. Early detection of malignant melanoma: The role of physician examination and self-examination of the skin. *Ca-A Cancer J Clinicians*, 35:130–151, 1985.
- [8] E. Golobardes, X. Llorca, M. Salamo, and J. Martí. Computer Aided Diagnosis with Case-Based Reasoning and Genetic Algorithms. In *Journal of Knowledge-Based Systems*, volume 15, pages 45–52, 2002.
- [9] R.M. Gutiérrez and N. Cortés. Confronting melanoma in the 21st century. In *Med Cutan Iber Lat Am*, pages 35(1):3–13, 2007.
- [10] J. Han and M. Kamber. Data mining: Concepts and techniques, 2006.
- [11] J. Handl, J. Knowles, and D.B. Kell. Computational cluster validation in post-genomic data analysis. *Bioinformatics*, 21(15):3201–3212, 2005.
- [12] L. Kaufman and P.J. Rousseeuw. Clustering by means of medoids. In *Statistical Data Analysis Based on the L1-Norm and Related Methods*, pages 405–416, North-Holland, 1987. Y. Dodge.
- [13] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability. Vol. 1.*, pages 281–297, 1967.
- [14] S. Montani, L. Portinale, G. Leonardi, and R. Bellazzi. Case-based retrieval to support the treatment of end stage renal failure patients. *Artificial Intelligence in Medicine*, 3:31–42, 2006.
- [15] P. Stefano, G. Fabbrocini, M. Scalvenzi, B. Emanuela, and M. Pensabene. Malignant melanoma clustering with some familiar/hereditary tumors. *Annals of Oncology*, 3:100, 2002.
- [16] D. Randall Wilson and Tony R. Martinez. Improved heterogeneous distance functions. *Journal of Artificial Intelligence Research (JAIR)*, 1:1–34, 1997.

Reasoning About Plans, Processes, and Actions

This page intentionally left blank

Using ant colony systems with pheromone dispersion in the traveling salesman problem

José Carlos BECCENERI^a and Sandra SANDRI^{b,1} and E.F. Pacheco da LUZ^a

^a LAC/INPE, Brazil

^b IIIA-CSIC, Spain

Abstract. Here we investigate the use of Ant Colony Systems (ACS) for the Traveling Salesman Problem (TSP). We propose the use of a modified ACS for graph problems in which we allow an artificial ant to lay down pheromone, not only on the edges in its path, but also on edges close to it: the closer a given edge is to one of those in the ant's path, the more pheromone it receives. The notion of edge closeness in the TSP takes into account not only the closeness among the nodes composing the edges but also the edges orientation.

Keywords. Fuzzy Sets theory, ant colony systems, traveling salesman, TSP

1. Introduction

An *Ant Colony System* (ACS) is an optimization method that employs a metaheuristic based on the collective behaviour of ants moving between the nest and a food source [3,2,4]. Each ant marks its path with a certain amount of pheromone and the marked path is further employed by other ants as a reference.

ACSs have been used successfully to solve several graph problems [4]. In most implementations of ACS for graph problems, an ant lays down pheromone only on the edges connecting the nodes in its path. However, to be more realistic, the pheromone should be modeled as an odor exhaling substance and, as such, the closest an ant would be to a trail of pheromone, the stronger should the perceived odor be. To mimic that characteristic, we proposed in [8] the use of ACS with pheromone dispersion, called *ACS-d* for short. In this approach, an ant is allowed to deposit pheromone not only on the edges in its path, but also on edges close to them. One of the main issues in this approach is thus to define, for each problem, how close a particular edge in the graph is to any of the edges composing a given path.

Notwithstanding the real world analogy, the main goal of modeling diffusion of pheromone is to allow a more efficient exploration of a problem's state space. We implemented the ACS-d approach in accordance to the use of a gradual fuzzy rule such as:

¹Correspondence to: Sandra Sandri, IIIA, Artificial Intelligence Research Institute CSIC, Spanish National Research Council Campus UAB, 08193 Bellaterra, Spain. Tel.: +34 93 580 9570; Fax: +34 93 580 9661; E-mail: sandri@iia.csic.es.

given an ant, “the closer is an edge to that ant’s path, the more pheromone shall it receive” [5]. The work in [8] is focused on the use of ACS-d to the function optimization problem, which was transformed as a graph problem in [1]. In this case, the notion of closeness between edges in the graph is simply modeled by the closeness between two possible values for an argument (the closeness between nodes and/or edges does not play a role).

In the present work we study the implementation of ACS-d’s for the *Traveling Salesman Problem* (TSP). In this problem, a salesman has to find a route which visits each of a set of cities once and only once (except for the starting one), such that the total distance travelled is minimal. This class of problems is usually modeled by an ACS in the following manner. Each city to be visited by a traveling salesman is associated to a node in a directed graph. Starting at a given node, each ant in a given generation produces a complete path connecting the set of cities such that each node has exactly one entrance edge and one exit edge. In the usual implementation, the ant associated to the shortest path among all ants, the “winning” or “leader” ant, is allowed to deposit pheromone on the edges composing its path [4]. To model the notion of closeness between two given edges in the TSP context, in the ACS-d we take into consideration not only the closeness between the nodes composing those edges but also the edges orientation.

The amount of pheromone deposited on an edge by a given ant is usually modeled by a constant. In the present work we examine the case in which the amount of pheromone that an ant lays down is proportional to the quality of its path, in accordance to the rule “the better is an ant’s path, the more pheromone shall that path receive”. We compare the use of ACS and ACS-d using the standard scheme (constant) with the variable (proportional) scheme.

This work is organized as follows. In the next section we briefly describe some notions on the basic type of ACS considered in this work, fixing the notation used in the remaining of the paper. Then in Section 3 we discuss the implementation of our approach for the traveling salesman problem. In Section 4 we describe a group of experiments and finally Section 5 brings the conclusion.

2. Some basic notions in ant colony systems and the traveling salesman problem

To solve a problem, an ACS is associated to a graph $G = (N, E)$ where N is a set of nodes and E is the complete set of edges that can be derived from N . A solution is a path linking a set of nodes obeying the restrictions imposed by the problem. At each generation, a set of n_a artificial ants generate a solution each. The ACS used here is such that only the best ant (according to a problem dependent criterium) is allowed to deposit pheromone on its path.

At the t -th iteration, the amount of pheromone on any edge $(i, j) \in E$ is given by

$$\tau_{ij}(t) = (1 - \rho) \times \tau_{ij}(t - 1) + \Delta\tau_{ij}(t) \quad (1)$$

where $\rho \in (0, 1]$ represents the pheromone decay rate; the pheromone increment $\Delta\tau_{ij}(t)$ is implementation dependent and shall be discussed later.

Usually, the next edge on an ant’s path is selected by a simple roulette scheme. The probability that an available edge $(i, j) \in E$ has of being chosen is calculated according to the formula

$$P_{ij}(t) = \frac{[\tau_{ij}(t)]^\alpha [\eta_{ij}]^\beta}{\sum_{l \in U} [\tau_{il}(t)]^\alpha [\eta_{il}]^\beta}, \quad (2)$$

where U is the set of nodes still unvisited by that ant and η_{ij} denotes the visibility/cost of each path, a concept that arises from the TSP, where the cost is the inverse of the distance of a particular path. Parameters α and β are weights that establish the tradeoff between the influence of accumulated pheromone and visibility in the calculation of the probability of an edge being chosen.

In another more complex roulette scheme, a random number in the range $[0, 1]$ is generated for an ant, which is compared with a parameter q_0 chosen for the problem. If the random number is greater than this parameter, the path is taken according to P_{ij} ; if not, the most marked path is chosen.

The traveling salesman problem can be described as follows. Given a set of n_c cities, a salesman has to find a route which visits each of the cities once and only once (except for the starting one), such that the total distance travelled is minimal. Formally, a solution consists in finding the minimal length Hamiltonian circuit on a graph $G = (N, E)$, where each node in N is associated to a city ($|N| = n_c$). The total length of a route is obtained by adding up the lengths associated to each of the edges in the path corresponding to that route.

Using an ACS, the problem is solved by creating generations of artificial ants that complete tours on the cities graph. In each iteration, after the completion of the tour, an ant a_k lays a quantity of pheromone on each edge (i, j) in its path. In the usual implementation, at iteration t we have:

$$\Delta\tau_{ij}^k(t) = \begin{cases} \tau_0, & \text{if } (i, j) \in T_k(t) \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

where $T_k(t)$ is the path taken by ant a_k at iteration t and τ_0 is a fixed increment.

Two schemes are usually employed to determine which ant is allowed to deposit pheromone on a iteration. In the first scheme, every ant is allowed to lay down pheromone on its path. Here we employ the second approach, in which only the best ant in an iteration, the “winning” or “leader” ant, is allowed to deposit pheromone on its path.

3. ACS with pheromone dispersion and the traveling salesman problem

In the ACS-d approach, an ant lays down a given amount of pheromone on the edges that compose its path, and lays down smaller quantities of that amount on the edges close to the ones on its path. Figure 1 illustrates pheromone deposition: edge (A, B) receives the full increment whereas nearby edges, departing from A , receive lesser amounts of it. In the following we discuss how we based our notion of closeness between edges for the TSP and how we formalized it.

3.1. Closeness between edges in the TSP

Let us consider a TSP with 11 cities, modeled by a graph with nodes $\{O, A, B, C, D, E, F, G, H, I, J\}$.

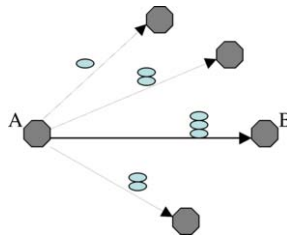


Figure 1. Example of pheromone dispersion in the TSP.

Let us suppose that an ant started its path on city O , then chose city A to go from O , and has just chosen city B to go from A . At this moment, the path is thus formed by edges (O, A) and (A, B) . In an ACS, at this point the ant would have deposited pheromone on edge (O, A) and would be about to deposit pheromone on edge (A, B) . This situation is illustrated in Figure 2.a.

In an ACS-d, the ant would now deposit pheromone not only on (A, B) but also on some edges close-by, all starting from A . Figure 2.b shows all the possible edges on which some pheromone could be dispersed.

The main point in our approach is to determine which edges starting on A should receive that extra amount of pheromone. In other words, how should we define a metric to measure closeness between edges ?

Before defining closeness between edges, we have to define closeness between nodes. In the following we shall consider that two nodes are somehow close if the Euclidean distance between them is smaller than a given threshold.

A first alternative to order the edges in the graph according to their closeness to edge (A, B) in Figure 1 is to take the nodes $r \in \{C, D, E, F, G, H, I, J\}$ and deposit the more pheromone on an edge (A, r) , the closest A is to r . Therefore, in this case, edges (A, C) and (A, D) would receive more pheromone than edges (A, E) and (A, F) (see Figure 3.a). However, this is counterintuitive, since (A, E) and (A, F) go somewhat in the same direction of (A, B) , whereas (A, C) and (A, D) do not.

Here we adopted a second alternative to model edge closeness: an edge (A, r) is the closest to an edge (B, r) , the closest node B is to node r (in terms of the Euclidean distance). Using this alternative, edges such (A, E) , (A, F) and (A, G) would be considered to be close to (A, B) and would be allowed to receive more pheromone than edges (A, C) and (A, D) (see Figure 3.b).

To limit vicinity, we have chosen to only accept edges (A, r) such that the distance between r and B is not greater than a threshold value d_{max} . For instance, if d_{max} is taken

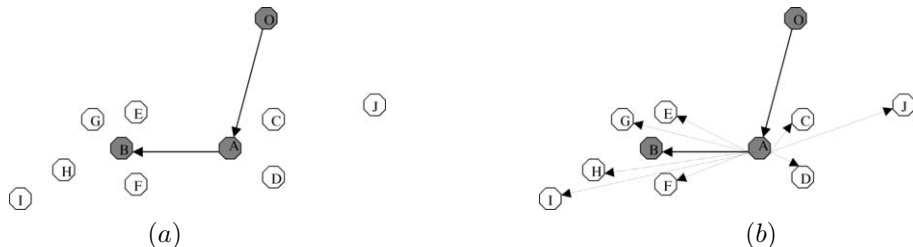


Figure 2. Edges that can possibly receive pheromone on the TSP: (a) in an ACS and (b) in an ACS-d.

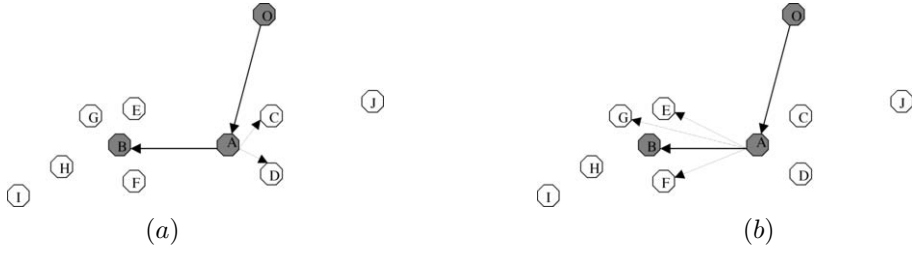


Figure 3. Strategies for the deposition of pheromone in the TSP by an ACS-d.

as the Euclidean distance between A and B , edges such as (A, E) , (A, F) and (A, G) would be allowed to receive extra pheromone, whereas edges (A, C) and (A, D) would not receive any.

Other strategies also seem to be worth considering. One could, for instance, choose the edges that are somewhat in the vicinity of both A and B . Applying this strategy in our example, edge (A, G) would not be allowed to receive any extra pheromone.

3.2. Standard versus proportional increment of pheromone

In the experiments shown in this work, we compare our approach ACS-d against an ACS, taking into account two types of pheromone deposition: the standard one and a proportional one:

- in the *standard* scheme, the initial amount of pheromone deposited on each edge belonging to a given ant's path is a fixed value for τ_0 , calculated as τ_0

$$\tau_0 = \frac{1}{n_c * Q}, \quad (4)$$

where Q is the tour length obtained by solving the problem with the greedy heuristics.

- in the *proportional* scheme, the amount of pheromone deposited on each edge belonging to a given ant's path is proportional to the quality of the path, and is calculated as

$$\tau_p = \tau_0 \times \frac{Q}{best_cost_turn} = \frac{1}{n_c * best_cost_turn}, \quad (5)$$

where Q is the length of the path obtained through the greedy heuristics and $best_cost_turn$ is the size of the path of the best ant in a turn.

Let $d(i, j)$ denote the Euclidean distance between two nodes i and j . Let $C_{i,j}$ be the set of edges considered to be close to (i, j) , defined as

$$C_{i,j} = \{(i, l) \in E \mid d(j, l) \leq d_{max}\},$$

where d_{max} is a constant (see 3.1).

Let us suppose that at the end of iteration t , an ant a_k is allowed to deposit pheromone on its path. The edges in the graph are updated as follows:

- each edge (i, j) in the path of ant a_k is updated by τ_0 , using formulas (4) or (5) depending on the type of pheromone deposition (standard or proportional);
- each edge $(i, l) \in C_{i,j}$ is updated by τ_0 multiplied by a reduction factor that implements the pheromone dispersion.

There are several possibilities for the reduction factor. Some of them are:

- a constant value $\gamma \in [0, 1]$, that stands for the maximal percentage of pheromone to be deposited on edges close to a winning ant's path. For example, with $\gamma = .2$, even the closest neighbor will receive at most 20% of the pheromone that was deposited on the edges belonging to the path itself.
- a value that depends on the closeness of a given edge to the best ant's path; the closest is a neighbor edge, the more pheromone it receives. For this purpose we can use a fixed value θ_k that is assigned to the k -th closest edge in the neighborhood of a given edge. In a neighborhood of size 3, we could use for example have $\theta_1 = .6, \theta_2 = .4, \theta_3 = .2$. We can also use a value that is proportional to the distance between node l and node j , considering that edge (i, j) is in the path and edge (i, l) is one of the neighbors of (i, j) that is allowed to receive a bit of extra pheromone. One such function is for example $\frac{1}{d(j,l)}$.
- a value that depends on the size of an edge: the smaller the edge, the more pheromone it receives.
- a value that is proportional to the quality of the best ant's path, as for example $\frac{Q}{best_cost_turn}$, where Q is the length of the path obtained through the greedy heuristics and $best_cost_turn$ is the size of the path of the best ant in a turn.

Here, the edges $(i, l) \in C_{i,j}$ are updated using the reduction factor as follows:

- in *standard scheme* we have

$$g_0 = \gamma \times \frac{1}{d(j,l)} \quad (6)$$

- in the *proportional scheme* we have

$$g_p = \gamma \times \frac{1}{d(j,l)} \times \frac{Q}{best_cost_turn} \quad (7)$$

4. Experiments

We implemented a simple ACS and an ACS-d and applied them on problems taken from [6]. Each problem is identified by a sequence of letters followed by a number that indicates the number of cities in the problem. The statistics were made using 10 different seed values for the random function and the edges were chosen according to the complex roulette scheme.

In order to distribute pheromone on edges close to a path in an ACS-d, we need to somehow keep a table of distances between nodes. To keep the complete table is clearly unfeasible as the number of cities grow. In the experiments described in the following, we have adopted the solution of keeping only the list of the closest h neighbors of each node, calculated as

$$h = \left\lceil \frac{n_c}{3} \times \log_{10} n_c \right\rceil.$$

A minimal of 6 neighbors is however imposed.

The parameters used in the experiments, unless stated otherwise, are:

- $\alpha = 1, \beta = 2, q_0 = 0.2, \rho = 0.2, \gamma = 0.2$

In each experiment, the number of ants is the same as the number of cities in each problem. The algorithm stop when a maximal number of iterations is reached. The increment τ_0 and the reduction factor are calculated as described above, considering the standard and the proportional schemes.

In each experiment, the maximal distance d_{max} was calculated as follows. We first put the complete set of distinct (non-oriented) edges of the graph in increasing order. Then we extracted half of the set, consisting of the shortest edges. Finally, we applied the arithmetic mean on the sizes of the edges in the selected set.

Note that, although the edges in a solution are oriented, pheromone is deposited on the edges without considering their orientation.

4.1. *Pheromone deposition: standard versus proportional schemes for a chosen experiment*

In Table 1, we detail the (truncated) results obtained for case *u52* from a traveling salesman library [6], varying the seed values used to generate the random numbers. We compared the results for the standard (s) versus proportional (p) scheme, and with (ACS) versus without pheromone dispersion (ACS-d).

Table 1. Cost in experiment *u52*: different seed values for random number generation.

| | s/p | 7 | 29 | 47 | 342 | 631 | 1111 | 1415 | 1689 | 3991 | 6618 |
|-------|-----|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| ACS | s | 9774 | 10008 | 10748 | 10552 | 10907 | 10237 | 9333 | 10495 | 10252 | 10139 |
| ACS | p | 10311 | 10581 | 10897 | 10861 | 9487 | 9894 | 10107 | 10119 | 10459 | 9557 |
| ACS-d | s | 10590 | 10599 | 10503 | 10656 | 10508 | 10674 | 10224 | 9775 | 10330 | 10362 |
| ACS-d | p | 7962 | 7742 | 8299 | 7980 | 7756 | 8067 | 7962 | 7927 | 7746 | 8109 |

Table 2. Cost in experiment *u52*: standard \times proportional schemes.

| | s/p | mean | std | min | max | cv |
|-------|-----|-------|-----|------|-------|-------|
| ACS | s | 10245 | 444 | 9334 | 10907 | 0.043 |
| ACS | p | 10228 | 466 | 9487 | 10897 | 0.04 |
| ACS-d | s | 10425 | 259 | 9775 | 10674 | 0.025 |
| ACS-d | p | 7955 | 169 | 7743 | 8300 | 0,02 |

Tables 1 and 2 show that the use of pheromone dispersion with the proportional scheme was superior than the other possibilities. A similar behavior was found in the other experiments, listed in 4.2; the best results have been obtained by ACS-d with the proportional scheme, and in relation to the simple ACS (without pheromone dispersion), the proportional scheme fared slightly better than the standard. For this reason, in the following, we present only the results for the proportional scheme, both for ACS and ACS-d.

4.2. Experiments with use of proportionality based on quality

In the following, we compare the ACS and ACS-d approaches on experiments in which the basic amount of pheromone deposited on each path is proportional to the quality of the solution associated to that path (scheme p). Table 3 (respec. 4) brings statistics on the costs (respec. mean number of iterations) for a set of problems from [6].

Table 3. Comparison between proportional ACS and proportional ACS-d: cost.

| case | ACS | | | | | ACS-d | | | | |
|---------|--------|------|--------|--------|------|--------|------|--------|--------|------|
| | mean | std | min | max | cv | mean | std | min | max | cv |
| u52 | 10228 | 466 | 9487 | 10897 | 0,04 | 7955 | 169 | 7743 | 8300 | 0,02 |
| u70 | 1011 | 45 | 920 | 1074 | 0,04 | 694 | 7 | 684 | 710 | 0,01 |
| rd100 | 14267 | 631 | 13193 | 15017 | 0,04 | 8266 | 246 | 7955 | 8934 | 0,03 |
| u120 | 2793 | 71 | 2657 | 2914 | 0,02 | 1655 | 20 | 1636 | 1707 | 0,01 |
| bier127 | 199655 | 7237 | 188401 | 211787 | 0,03 | 125720 | 2008 | 122779 | 129530 | 0,02 |
| ch130 | 121189 | 443 | 11583 | 12793 | 0,00 | 6576 | 174 | 6342 | 6983 | 0,03 |
| d198 | 31918 | 1470 | 28620 | 34000 | 0,04 | 16478 | 246 | 16031 | 16842 | 0,02 |
| gil262 | 6659 | 239 | 6199 | 6923 | 0,03 | 2754 | 119 | 2564 | 3001 | 0,04 |

Table 4. Comparison between proportional ACS and proportional ACS-d: mean number of iterations.

| case | max | ACS | ACS-d |
|---------|-------|------|-------|
| u52 | 10000 | 696 | 3334 |
| u70 | 10000 | 905 | 3325 |
| rd100 | 10000 | 1053 | 4027 |
| u120 | 8500 | 1260 | 3964 |
| bier127 | 4000 | 1387 | 3350 |
| ch130 | 2000 | 987 | 1774 |
| d198 | 2000 | 1285 | 1803 |
| gil262 | 1000 | 943 | 988 |

We see that ACS-d with the proportional scheme has performed better in all experiments in terms of cost, ACS-d yielding from 5% to 50% lower cost than ACS. On the other hand, the best solution took more iterations to be found in ACS-d than ACS. In particular, inside each experiment very often a number of iterations close to the maximum was reached (2500), something that seldom occurred with the simple ACS. This suggests that ACS-d takes longer to converge and is better at avoiding local minima. Nevertheless, more experiments are needed varying the number of neighbors and dispersion rate γ to see whether more efficiency can be achieved with ACS-d.

5. Conclusions

We presented a modified approach to ACSs in graph problems, in which small amounts of pheromone are laid down on edges that are close to those composing the winning ant's path. The notion of closeness changes according to the problem and here we have dis-

cussed how to implement this notion in the particular case of the traveling salesman problem. We verified the use of pheromone dispersion in two schemes for laying pheromone on the edges on a path: one that uses a fixed amount of pheromone τ_0 and one in which the amount deposited is proportional to the quality of the solution.

Even though the experiments we have made so far are not exhaustive, the good results we obtained with many instances of the TSP encourage us to investigate further in this direction. In the future, we intend to apply the ACS-d approach with the proportional scheme to solve the TSP with a large number of cities. Also, we intend to investigate how to obtain good results with this approach yet avoiding unreasonable computing times both with parallelization and with suitable data structures.

Acknowledgements

This work was partially funded by IEA (TIN2006-15662-C02-01), OK (IST-4-027253-STP), eREP (EC-FP6-CIT5-28575) and Agreement Technologies (CONSOLIDER CSD2007- 0022, INGENIO 2010).

References

- [1] J.C. Becceneri and A.S.I Zinober. Extraction of energy in a nuclear reactor by ants. Proc. of the 33th Brazilian Symposium of Operational Research (SOBRAPO'01), Campos do Jordão (Br), 2001.
- [2] E. Bonabeau, M. Dorigo and G. Theraulaz. From natural to artificial swarm intelligence. Oxford University Press, 1999.
- [3] M. Dorigo, V. Maniezzo and A. Coloni. The ant system: optimization by a colony of cooperating agents. IEEE Trans. SMC-PartB, Vol. 26, No. 1, 1996, pp. 1-13.
- [4] M. Dorigo, G. Di Caro and L.M. Gambardella. Ant algorithms for discrete optimization. Artificial Life, Vol. 5, No. 3, pp. 137-172, 1999.
- [5] D. Dubois and H. Prade. What are fuzzy rules and how to use them. Fuzzy Sets and Systems, Vol. 84, No. 2, pp. 169-185, 1996.
- [6] <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>
- [7] R. Souto, H. Campos Velho, S. Stephany, S. Sandri. Reconstruction of Chlorophyll Concentration Profile in Offshore Ocean Water using a Parallel Ant Colony Code 1st Int. Workshop on Hybrid Metaheuristics (HM2004/ECAI'04), Valencia (Es), 2004.
- [8] J.C. Becceneri and S. Sandri. Function optimization using ant colony systems. IPMU'06, Paris (Fr), 2006.

Building Policies for Scrabble

Alejandro GONZALEZ ROMERO and René ALQUEZAR

Departament de Llenguatges i Sistemes Informàtics

Universitat Politècnica de Catalunya,

C/ Jordi Girona, 1-3, Edifici Omega, 08034 Barcelona – Spain

yarnalito@gmail.com, alquezar@lsi.upc.edu

Abstract. Research to learn policies using Evolutionary Algorithms along with training examples has been done for the domains of the Blocks World and the KRKa2 chess ending in our previous work [1,2]. Although the results have been positive, we believe that a more challenging domain is necessary to test the performance of this technique. The game of Scrabble, played in Spanish, in its competitive form (one vs. one) intends to be used and studied to test how good evolutionary techniques perform in building policies that produce a plan. To conduct proper research for Scrabble a Spanish lexicon was built and a heuristic function mainly based on probabilistic leaves was developed recently [3]. Despite the good results obtained with this heuristic function, the experimental games played showed that there is much room for improvement. In this paper a sketch of how can policies be built for the domain of Scrabble is presented; these policies are constructed using attributes (*concepts* and *actions*) given by a Scrabble expert player and using the heuristic function presented in [3] as one of the *actions*. Then to evaluate the policies a set of training examples given by a Scrabble expert is used along with the evolutionary learning algorithm presented in [2]. The final result of the process is an ordered set of rules (a policy) which denotes a plan that can be followed by a Scrabble engine to play Scrabble. This plan would also give useful information to construct plans that can be followed by humans when playing Scrabble. Most of this work is still under construction and just a sketch is presented. We believe that the domain of games is well-suited for testing these ideas in planning.

Keywords. Computer games, Scrabble, learning policies, planning, heuristic function, evolutionary algorithms

Introduction

The game of Scrabble is a good platform for testing Artificial Intelligence techniques. The fact that Scrabble has a significant random component makes it very interesting for AI research. Poker and bridge are also interesting games with hidden information. Significant progress has been made in managing the hidden information in those games and in creating computer agents that can compete with intermediate-level human players [4,5].

The game has an active competitive tournament scene, with national and international Scrabble associations and an annual world championship for Spanish language players.

Tournament Scrabble is a 2-player game where each player has a rack of seven lettered tiles that are randomly drawn from a bag that initially contains 100 tiles. Players take turns put on the board tiles across or down in crossword fashion forming

words that must appear in an official dictionary or lexicon. Besides playing on the board, a player can choose to change tiles or pass in the turn. The winner of the game is the player that achieves the highest score after all tiles are played.

To obtain high scoring moves, it is convenient to use the premium squares and to play *bingos*. A **bingo** is a move where all seven tiles of the rack are played at once. This move awards an extra 50 points bonus plus the score of your word on the board. An approximated average score of a move without bingos is 20 points. An intuitive idea to maximize the amounts of points of a player is to play as many bingos as possible.

Therefore it is convenient to have Scrabble *leaves*, the tiles remaining in the rack after a move, as close to a bingo as possible. This was taken into account in [3] to build a heuristic function mainly based on maximizing the probability of a bingo in the next move. In fact this heuristic function tries to make a balance between maximizing one's score in the present move and maximizing the probability of a bingo in the next move. In Scrabble, "achieving a high score requires a delicate balance between maximizing one's score on the present turn and managing one's rack in order to achieve high-scoring plays in the future" [6].

Even though Scrabble is already dominated by computer agents [7, 8], there is still room for improvement in Computer Scrabble. For instance, there is not yet Scrabble software capable of generating policy sets (policies) that yield Scrabble plans. Many ideas in these plans could be followed by humans to improve their game.

Despite the good results obtained with our heuristic function presented in [3], the experimental games played showed that much improvement can be made. In this paper policies are built for the domain of Scrabble; they are constructed using attributes given by a Scrabble expert player and the heuristic function presented in [3]. Then to evaluate the policies, a set of training examples given by a Scrabble expert is used along with the evolutionary learning algorithm presented in [2]. The final result of the process is an ordered set of rules (a policy) which denotes a plan that can be followed by a Scrabble engine to play Scrabble.

1. About the Scrabble Tools

1.1. The Lexicon

An important part to conduct research in Scrabble, in Spanish for our case, is the construction of a lexicon. The length of it is significantly larger than the length of a corresponding lexicon in English. The total number of words (of length less than 16) is, approximately, 246000 in SOWPODS and 635000 in our Spanish lexicon.

The electronic version of DRAE (Diccionario de la Real Academia Española), the official Scrabble dictionary in Spanish has approximately 83000 lemmas (entries) from which one has to construct the complete lexicon.

A free English Scrabble source program made by Amitabh [9] was modified to obtain Scrabler II, a Spanish Scrabble program that plays with the official rules of Spanish Scrabble tournaments (the FISE rules). The lexicon was tested using this Scrabble program. The lexicon has also been tested in Spanish tournaments like the past 2008 Scrabble National Championship of Spain. It was used to play duplicate Scrabble (a variant of Scrabble where everyone has the same tiles each turn, and the

best move is the highest scoring move in each turn). Another Scrabble program named TimWi, with the aid of the lexicon, was used for this purpose. More details about the lexicon are given in [3].

1.2. The Heuristic Function

A heuristic function mainly based on probabilities that could guide a Scrabble engine was presented in [3]. This heuristic function tries to make a balance between maximizing one's score in the present move and maximizing the probability of a bingo in the next move. We recall here this heuristic evaluation function:

$$v = j + p \cdot b - d \quad (1)$$

where j is the number of points made by the move, in which t tiles are played¹; $j=0$ if the t tiles are changed rather than played on the board; p is the probability of obtaining a 7-letter bingo if t tiles are drawn at random from a set that is the union of the bag and the opponent's rack (which we call the "augmented bag"); b is the expected value of a bingo, which can be estimated as 77 (an average taken from 1000 games) or better²

$$b = 50 + 2.5 (r + 1.92t) \quad (2)$$

where t is the number of tiles drawn from the bag and r is the total number of points of the residue (=rack leave); d is a nonnegative number which is zero if the move is not weak from a defensive point of view. The value of d is 20, say, if one puts a tile on the edge allowing a possible nine-timer; it is 10 if one plays a vowel next to a premium square allowing a six-timer; it is 5 if one plays allowing a possible four-timer.

2. Learning the Policies (research work)

2.1. Rules and Policies

A training example for the Scrabble domain would consist of a certain Scrabble position, the tiles of the player whose turn is, the number of tiles left in the bag, the current score of the game, and a good move given by an expert Scrabble player. It is also possible to have a set of good moves instead of only one good move. A Scrabble training example is presented later in this section (see subsection 2.3).

To represent rules about our domain a set of numbered predicates is used; it is divided into 2 subsets: the first set contains *concepts* and the second one is conformed of *actions*.

A rule is formed of two parts: its LHS (Left Hand Side) is a conjunction of predicates which will be called *concepts*, and its RHS (Right Hand Side) is a set

¹ We assume here that the bag has at least t tiles.

² Explanations of b and p are given in [3]

consisting of a single predicate which will be called an *action*. The negation of a concept is itself a concept. For instance in the blocks world we could have rules like:

$$\pi_1 = clear_s(X) \wedge \neg on_table_s(X) \rightarrow move(X, table)$$

which can be read as “if the block X is clear and it is not on the table then move X to the table”.

Using the training examples the performance of every single rule is tested and a *partial fitness* is assigned to it in the way described in Algorithm 1. To construct policies, sets of rules are generated; these sets are called *policies* or *candidates*. The *final fitness* of a candidate is calculated by adding the partial fitness of all its rules.

We now give some definitions specially modified for the Scrabble domain:

Definition 1. A rule R satisfies a training example E if and only the LHS (left hand side) of the rule R holds (remains true) on the Scrabble position given in E .

Definition 2. Suppose rule R satisfies the training example E . Then if the RHS (right hand side) of the rule, the action, produces a move from the good moves set of E then R correctly satisfies E ; otherwise R incorrectly satisfies E .

To show how these ideas would work for the Scrabble domain, let us build a simple policy for Scrabble. But first let us define a few things.

Definition 3. Let a *bingo line* be a place on the board where a bingo can be inserted.

Definition 4. Let us call an *open board* a board position in which there is at least one bingo line; and a *closed board* a board position in which there are no bingo lines.

Table 1. Scrabble concepts

| No. | Concepts | Meaning of Concepts |
|-----|-----------------------|--|
| 1. | <i>Middlegame(X)</i> | X is a Middlegame position |
| 2. | <i>Pre_endgame(X)</i> | X is a Pre_endgame position |
| 3. | <i>Endgame(X)</i> | X is an Endgame position |
| 4. | <i>Openboard(X)</i> | X is an Open board position |
| 5. | <i>Closedboard(X)</i> | X is a Closed board position |
| 6. | <i>Slightad</i> | The current player has a slight advantage ($0 < ad \leq 50$) |
| 7. | <i>Mediumad</i> | The current player has a medium advantage ($50 < ad < 100$) |
| 8. | <i>Somead</i> | The current player has some advantage ($ad \geq 100$) |
| 9. | <i>Slightdis</i> | The current player has a slight disadvantage ($0 < dis \leq 50$) |
| 10. | <i>Mediumdis</i> | The current player has a medium disadvantage ($50 < dis < 100$) |
| 11. | <i>Somedis</i> | The current player has some disadvantage ($dis \geq 100$) |
| 12. | <i>Q_notplayed(X)</i> | The letter Q has not been played in position X |
| 13. | <i>Q_in_the_rack</i> | The letter Q is in the rack |
| 14. | <i>U_in_the_rack</i> | The letter U is in the rack |
| 15. | <i>Opp_Passed</i> | The opponent has just passed in the last move |
| 16. | <i>Opp_Played_1T</i> | The opponent has just played one Tile in the last move |

Definition 5. Let t denote the number of tiles inside the bag. If a position is such that:

| | |
|-----------------|--|
| $t > 14$ | then let this position be a <i>middlegame</i> position. |
| $0 < t \leq 14$ | then let this position be a <i>pre_endgame</i> position. |
| $t = 0$ | then let this position be an <i>endgame</i> position. |

To build rules for Scrabble we will use 30 predicates, a set of 16 concepts and a set of 14 actions. Tables 1 and 2 give those sets.

Table 2. Scrabble actions

| No. | Actions | Meaning of Actions |
|-----|------------------------|---|
| 1. | <i>Move_Heuri</i> | The move is performed by the Heuri strategy defined in 1.2 |
| 2. | <i>Move_Montecarlo</i> | The move is performed using Montecarlo Simulation. |
| 3. | <i>Move_usingB*</i> | The move is performed using the B* algorithm of search. |
| 4. | <i>Move_Max</i> | Move the maximum scoring move |
| 5. | <i>Move_MaxDouble</i> | Max double move |
| 6. | <i>OpenMove</i> | Move that opens bingo lines or premium squares |
| 7. | <i>BlockMove</i> | Move that closes bingo lines or premium squares |
| 8. | <i>Pass</i> | Do not move and pass the turn to the opponent |
| 9. | <i>KeepU</i> | Keep the U tile in the rack (do not play the U tile) |
| 10. | <i>ChangeQ</i> | Change the Q tile |
| 11. | <i>ChangeHighT</i> | Change one or more high value tiles. |
| 12. | <i>KeepTonbag</i> | Keep at least one tile on the bag after the move |
| 13. | <i>Emptybag</i> | Empty the bag after the move (take all tiles inside the bag). |
| 14. | <i>KeepHighT</i> | Keep at least one high value tile in the rack after the move. |

We now explain briefly some of the above actions.

Move_Montecarlo: Simulation will be used to find the move to make in the current position. Simulation techniques take into account the possible replies to a candidate move, the replies to those replies, and so on for many plies. This method rests on Monte Carlo sampling for the generation of the opponent's rack, in situations with uncertainty, and it has its theoretical base in the Minimax technique of Game Theory. The programs Maven [7] and Quackle[10] are good references that exploit this technique.

*Move_usingB**: B star (B*) is a best-first graph search algorithm that finds the least-cost path from a given initial node to a goal node (out of one or more possible goals). The idea is to use B*-search [11] to tackle the endgame as Maven does. Recall that the endgame starts when the last tile is drawn from the bag.

Now we will define two actions: “*Move_Heuri*” and “*Move_MaxDouble*”. Let us name the heuristic function presented earlier as *Heuri*. And let us define a *Double* as a move consisting in playing twice in a row, imagining the opponent passes. Finally, define *MaxDouble* as the Double that gives the maximum score on the board (the score in this case is the sum of the individual scores of the moves involved in a Double). The order of the Double move will be to play always the highest move first.

Therefore the action “move_Heuri” would indicate that the move to be made is given by *Heuri*. And the action “move_MaxDouble” would indicate to perform the first move (also the highest) of a *MaxDouble*.

Another possible aspect in Scrabble strategy is to take into account the current score of the game. For instance, if a player is well behind in the score, a possible action would be to make “come back moves”, this type of moves would risk opening bingo lines, for example, to catch up. The action *OpenMove* serves for this purpose. On the other hand, if a player has a good lead, he/she might try to hold the lead by making “blocking moves”. A *blocking move* could be any move that reduces the number of bingo lines, or covers triple premium squares. The *ClosedMove* action has this idea.

Let X be any Scrabble board position, then two examples of policies are:

Policy I

$$\begin{aligned}
 \pi_1 &= \text{OpenBoard}(X) \wedge \text{Middlegame}(X) \rightarrow \text{Move_Heuri} \\
 \pi_2 &= \text{ClosedBoard}(X) \wedge \text{Middlegame}(X) \wedge \text{slightad} \rightarrow \text{Move_MonteCarlo} \\
 \pi_3 &= \text{ClosedBoard}(X) \wedge \text{Pre_endgame}(X) \wedge \text{Somedis} \rightarrow \text{OpenMove} \\
 \pi_4 &= \text{OpenBoard}(X) \wedge \text{Pre_endgame}(X) \wedge \text{Somead} \rightarrow \text{ClosedMove} \\
 \pi_5 &= \text{ClosedBoard}(X) \wedge \neg \text{Middlegame}(X) \rightarrow \text{move_MaxDouble}(X) \\
 \pi_6 &= \text{Pre_endgame}(X) \wedge \neg \text{Somead} \wedge U_in_the_rack \wedge Q_notplayed(X) \rightarrow \text{KeepU} \\
 \pi_7 &= \text{Mediumad} \wedge \text{Opp_Passed} \wedge Q_notplayed(X) \rightarrow \text{Pass} \\
 \pi_8 &= \text{Endgame}(X) \rightarrow \text{Move_usingB*}
 \end{aligned}$$

Policy II

$$\begin{aligned}
 \pi_1 &= \text{OpenBoard}(X) \wedge \text{Middlegame}(X) \rightarrow \text{Move_Heuri}(X) \\
 \pi_2 &= \text{ClosedBoard}(X) \wedge \neg \text{Middlegame}(X) \rightarrow \text{Move_MaxDouble}(X) \\
 \pi_3 &= \text{OpenBoard}(X) \wedge \neg \text{Middlegame}(X) \rightarrow \text{Move_Heuri}(X) \\
 \pi_4 &= \neg \text{OpenBoard}(X) \wedge \text{Middlegame}(X) \rightarrow \text{Move_Heuri}(X)
 \end{aligned}$$

Later in the article this policy II is explained using a training example.

2.2. The Learning Evolutionary Algorithm

Using the training examples the performance of every single rule is tested and a *partial fitness* is assigned to it in the way described in Algorithm 1. To construct plans, sets of rules are generated; these sets are referred as *candidates*, *policies* or *policy sets*. The *final fitness* of a candidate is calculated by adding the partial fitness of all its rules.

The system will learn a policy set of rules by using a generational evolutionary algorithm that consists of the following steps:

2.2.1. Seeding

In this step, the population is initialized by generating the initial policy sets of rules (i.e. the initial *candidates*). The order of the rules in a candidate is relevant, since for each

candidate, the rules are checked from the top rule to the bottom rule until a rule can be applied, in the way classical planners work.

2.2.2. Evaluating the fitness

To make better candidates the first thing that is needed is a way to judge how good a candidate is, i.e. the *fitness* of a candidate. Candidates are evaluated according to the following algorithm:

Let C denote a candidate, T a training example, r a rule, and let *stillRulesToCheck* and *fires* be two Boolean variables.

```

For  $C$ := the first candidate to the last candidate do
  Initialize the partial fitness of all the rules of  $C$  to zero;
  For  $T$ := the first training example to the last training example do
     $fires$ :=false;  $stillRulesToCheck$ :=true;
     $r$ := the first rule of  $C$ ;
    While not  $fires$  and  $stillRulesToCheck$  do
      If correctlySatisfies ( $r,T$ ) then
        Add one to the partial fitness of rule  $r$  ;
         $fires$ := true
      else if incorrectlySatisfies ( $r,T$ ) then
        Subtract ten from the partial fitness of rule  $r$  ;
         $fires$ := true
      else if  $r$  is the last rule of  $C$  then
         $stillRulesToCheck$ :=false
      else  $r$ := the next rule of  $C$ 
    endIf
  endWhile
endFor
  Compute the fitness of  $C$  as the sum of the partial fitness of all its rules
endFor

```

Algorithm 1. Evaluating the fitness of the candidates.

2.2.3. Breeding

Once the candidates are evaluated they are ordered according to their final fitness. Then to select the candidates that should evolve, a selection probability is used. This probability is an increasing function of the final fitness of each candidate. Once a pair of candidates is selected, a certain “crossover probability” is used to possibly perform a 1-point crossover on them, so forming two offspring. If no crossover takes place, two offspring are formed that are exact copies of their respective parents. Mutation is performed on the two offspring using a determined “mutation rate”.

After replacing the current candidate population with the new candidate population we re-evaluate each candidate going through the *Evaluating the fitness* step again.

This process continues until a certain fixed number of generations is reached or another stop criterion is satisfied.

2.3. A Scrabble Training Example and a Policy Example

A training example is given below, in Figure 1, to better explain some of the issues treated so far in the paper.

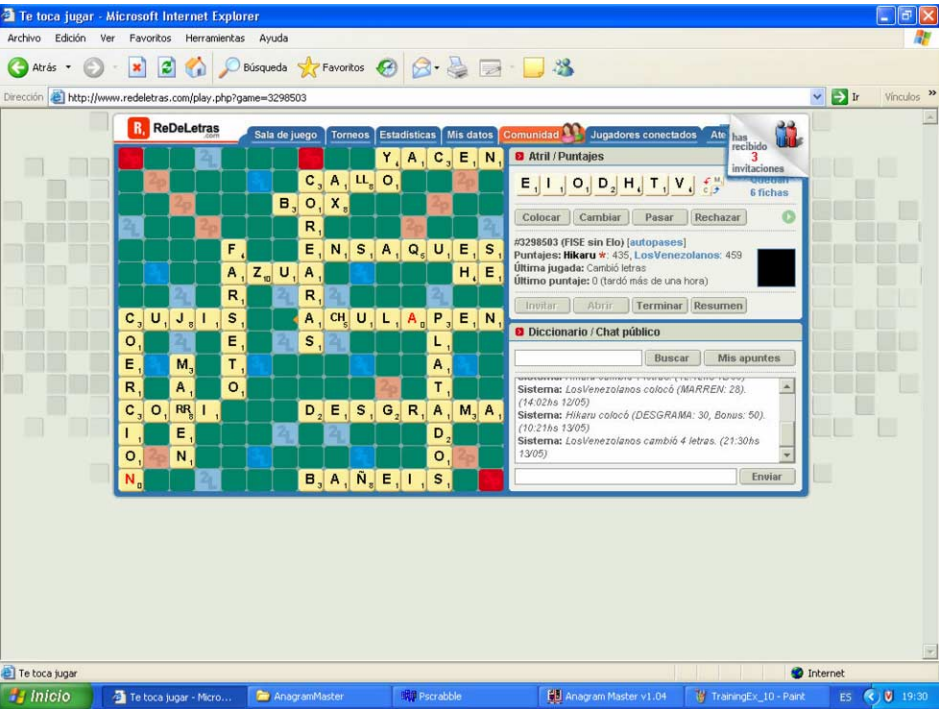


Figure 1. A training example (Good move (VADEO 15K V))

As can be seen in Figure 1, the training example consists in: a Scrabble position, the tiles of the player in turn (E,I,O,D,H,T,V), the number of tiles left on the bag (“Quedan 6 fichas” on the upper right corner (the total number of tiles is 100)), the current score of the game (Hikaru 435, LosVenezolanos 459), and a good move with its position on the board, VADEO 15K V (VADEO is placed vertically (V) starting at column 15 and row K), given by a Scrabble expert. (VADEO 15K V is vertically placed on the lower right corner, it touches a triple letter word and scores 27 points).

Let X be any Scrabble board position, then an example of a policy or candidate is:

Candidate 1

$$\pi_1 = \text{OpenBoard}(X) \wedge \text{Middlegame}(X) \rightarrow \text{Move_Heuri}$$

$$\pi_2 = \text{ClosedBoard}(X) \wedge \neg \text{Middlegame}(X) \rightarrow \text{Move_MaxDouble}$$

$$\pi_3 = \text{OpenBoard}(X) \wedge \neg \text{Middlegame}(X) \rightarrow \text{Move_Heuri}$$

$$\pi_4 = \neg \text{OpenBoard}(X) \wedge \text{Middlegame}(X) \rightarrow \text{Move_Heuri}$$

For simplicity the word position will be used from now on to denote a board position. Although now we can always replace the concept $\text{ClosedBoard}(X)$ by the concept $\neg \text{OpenBoard}(X)$, we defined them separately because in the future we might introduce a new $\text{SemiOpenBoard}(X)$ concept. *Candidate 1* can be read as follows:

π_1 If the position is open and it is a middlegame position then move using Heuri

π_2 If position is closed and it is not a middlegame position then move a MaxDouble

π_3 If the position is open and it is not a middlegame position then move using Heuri

π_4 If the position is not open and it is a middlegame position then move using Heuri

First we will analyze the rules one by one to clarify the definitions, and then we will run Algorithm 1 on this candidate.

Rule π_1 does not satisfy the training example, see Figure 1, since the OpenBoard concept of the LHS of the rule (the position is open) is not true in the training example. After analysing the training example, an expert player, this time helped by an anagram solver, determined that it is not possible to put a Bingo on the board with the remaining tiles. Therefore the position for the training example is not open, it is closed.

Rule π_2 does satisfy the training example in Figure 1, since the concepts of the LHS of the rule are both true (the position is closed and since there are only 6 tiles left, it is not a middlegame position). Besides, after careful analysis by the expert, there are two MaxDouble: 1) VADEO 15K V and HI M9 H (HI is placed horizontally (H) starting at row M and column 9, also forming EH and SI) and 2) VADEO 15K V and HI B14 H (HI is placed horizontally (H) starting at row B and column 14, also forming EH and NI). The action MaxDouble will produce the highest move VADEO 15K V. Therefore since it belongs to the set of good moves, rule π_2 correctly satisfies the training example.

Similarly rules π_3 and π_4 do not satisfy the training example.

When running *Algorithm 1* on the policy *Candidate 1* we will get that the only rule that scores is the rule π_2 , then we add one to its partial fitness. And since there are no more training examples and no more candidates or policies, the algorithm will stop and the fitness of *Candidate 1* will be one. *Candidate 1* can be read from top to bottom to give a plan or strategy to follow by a Scrabble engine or it can be semi-followed by a human.

3. Results, conclusions and future work

A short match between two Scrabble strategies was performed. The first strategy was to always used *Heuri* (the probabilistic heuristic Scrabble function), the second strategy (*Poli*) used the policy presented in this paper. The match consisted in 28 games and the result was Poli 16, Heuri 12. Poli seemed to play better the endings. This is due to the

special action `Move_Max_double`, this action many times works well in the late phases of the game where there is not much room to put bingos on the board and the tiles left inside the bag usually do not serve to make bingos. In the endgame a key goal is to use up all your tiles before the opponent does. Then your opponent will not increase his score, and you will have the additional benefit of the face value of his rack being added to your own score.

Although this is a very short match, it gives us hopes that our policy learning technique might work for the domain of Scrabble and perhaps for other domains that have hidden information. We believe that it is strongly convenient to build good concepts and actions, as well as key training examples to help the learning algorithm. It is also interesting to see if it can learn something using many random training examples.

Probably the phase of the game that needs more work is the `pre_endgame` phase where the goal is to reach a favourable endgame. Therefore it might be wise to construct and solve more training examples for this phase of the game.

One can also model the strategic component so as to reduce uncertainty, inferring information from the opponent's racks. For this purpose, the probabilistic models are especially adequate. Recently Richards and Amir [6] used Probability theory (Bayes theorem) to gain knowledge about the opponent tiles. In the future, we could include special predicates for Opponent Modelling using this approach.

Finally to test the policies built it is necessary to perform many matches, not only against humans, but against strong Scrabble engines like Maven [7] or Quackle[10]. These matches can take place in the Computer Olympiads that are held every year by the ICGA (International Computer Game Association).

References

- [1] González Romero A., Alquézar R. To block or not to block? In *Advances in Artificial Intelligence: Iberamia 2004*, Puebla, Mexico, Springer LNCS 3315, pp.134-143, 2004.
- [2] Alquézar R., Gonzales Camargo A., González Romero A. Evolving plans for the KRKa2 chess ending., *Proc. 24th IASTED Int. Conf. on Artificial Intelligence and Applications*, Innsbruck, Austria, pp.80-84, 2006.
- [3] González Romero A., González Acuña F., Ramírez Flores A., Roldán Aguilar A., Alquézar R., Hernández E. A heuristic for Scrabble based on probability, *Proc. ECAI'08 Workshop on Artificial Intelligence in Games*, Patras, Greece, 2008.
- [4] Billings D., Davidson A., Schaeffer J., and Szafron D. The challenge of poker. *Artificial Intelligence*, 134:201-240, 2002.
- [5] Ginsberg M.L., GIB: Steps toward an expert-level bridge-playing program. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence (IJCAI-99)*, pages 584-589, 1999.
- [6] Richards, M., Amir, E., "Opponent modeling in Scrabble", *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence*, 1482-1487, Hyderabad, India, January 2007.
- [7] Sheppard B. World-championship-caliber Scrabble. *Artificial Intelligence*, 134(1-2):241-275, 2002.
- [8] Sheppard B. Toward Perfection of Scrabble Play. PhD thesis, Computer Science, University of Maastricht, 2002.
- [9] Amitabh S., Scrabblor: award winning Scrabble program built as part of a programming competition at IIT Delhi, <http://www-personal.umich.edu/~amitabh/scrabble.html>, 1999.
- [10] Katz-Brown J., O'Laughlin J. et al. *Quackle* is an open source crossword game program released in March 2006, <http://web.mit.edu/jasonkb/www/quackle/>
- [11] Hans J. Berliner, The B* tree search algorithm; A best-first proof procedure, *Artificial Intelligence*, v.12 , p.23-40, 1979.

This page intentionally left blank

Robotics

This page intentionally left blank

Monocular object pose computation with the foveal-peripheral camera of the humanoid robot Armar-III¹

Guillem ALENYÀ^a Carme TORRAS^a

^a *Institut de robòtica i Informàtica Industrial (CSIC-UPC)*

Abstract. Active contour modelling is useful to fit non-textured objects, and algorithms have been developed to recover the motion of an object and its uncertainty. Here we show that these algorithms can be used also with point features matched in textured objects, and that active contours and point matches complement in a natural way. In the same manner we also show that depth-from-zoom algorithms, developed for zooming cameras, can be exploited also in the foveal-peripheral eye configuration present in the Armar-III humanoid robot.

Keywords. Active contours, motion estimation, depth from zoom

Introduction

Active contours, parameterised as B-splines, are interesting to extract object motion in dynamic scenes because they are easy to track, computation time is reduced when control points are used, and they are robust to partial occlusions. One important property is that a contour is easy to find, even in poorly textured scenes, where most of the point matching algorithms fail. We have been using contours in this way in several works [2,3,4] mainly to recover camera egomotion. One main concern is active contour initialisation. We have always considered that the contour was initialized by an operator. There are several works on contour fitting, but few on automatic contour initialization. Generally, the number of B-splines initialized is too high for general images [10], or either methods require very well segmented images [27].

Here we explore the possibility of using the algorithms developed for active contours with point matches instead. Contrarily to active contours, in order to find enough and reliable point correspondences the objects in which we put attention should be richly textured. As we can see, points and contours complement themselves in a natural way.

It is common to consider Sift features [19] as the state-of-the-art in point detection and matching. Sift features are considered to be invariant to changes in position and orientation, and to scale and illumination up to some degree. Unfortunately, the time required to process a frame, computing all sift features and comparing them with features

¹This research is partially funded by the EU PACO-PLUS project FP6-2004-IST-4-27657, the Consolider Ingenio 2010 project CSD2007-00018, and the Catalan Research Commission. G. Alenyà was supported by the CSIC under a Jae-Doc Fellowship.

in the model, is too long. Some alternatives have appeared, notably Surf [7] and Fast Keypoints [17]. In this work we aim to assess the feasibility of the developed active contour algorithms when control points are replaced by point matches, and computing time limitations are not taken into account. In the future, to apply this approach to real time scenarios a faster implementations should be developed.

Vision algorithms to recover object or camera motion need the information of the camera internal parameters and initial distance from the camera to the object to obtain metric information. Depth between the camera and the object is usually estimated using two cameras. Unfortunately, maintaining the calibration between both eyes in a lightweight active head, as that present in humanoid robots, is difficult, as head motions and especially saccadic motions produce slight eye motions that are sufficient to uncalibrate the stereo pair. This is the reason why 2D measures are sometimes preferred over 3D reconstruction data when controlling the gaze direction of an active head [26]. We will propose an alternative approach to recover initial depth, which makes use of two cameras with different focal lengths.

This paper is structured as follows. Section 1 presents the principles to compute 3D scene motion from a suitable parameterisation of the motion in the image. In Section 2 the algorithm to compute depth-from-zoom is described. Experiments are presented in Section 3. First, some experiments on depth recovery, and in Section 3.2 a metric reconstruction of an object motion from monocular image streams by making use of the depth recovered in the previous experiment. Finally, conclusions are drawn in Section 4.

1. Pose computation

Object pose involves 3 degrees of freedom (dof) for translation and 3 for orientation. Its computation from point matches is a classical problem in computer vision [12]. A lot of works deal with this problem, in affine viewing conditions [14,18,24] as well as in more general perspective conditions [8,13].

Assuming restricted affine imaging conditions instead of the more general perspective case is advantageous when perspective effects are not present or are minor [25]. The parameterisation of motion as an affine image deformation has been used before for active contour tracking [9], qualitative robot pose estimation [20] and visual servoing [11].

To obtain the pose of an object we use two different views of the same object. We use the first image as the initial position and we relate the motion present in the second image to the first one. If we assume affine imaging conditions, the deformation of the point cloud in the second image with respect to the initial one can be parameterised using a 6 dof *shape vector*, that codifies the possible affine deformations, independently of the number of point matches used. This is an advantageous representation because it is more compact and consequently operations to implement tracking algorithms are simpler.

Similarly as has been done in [4], we can consider extracted points \mathbf{Q} as control points of a B-spline, and build a *shape matrix* [20] of the form

$$\mathbf{W} = \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} \mathbf{Q}^x \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \mathbf{Q}^y \end{bmatrix}, \begin{bmatrix} 0 \\ \mathbf{Q}^x \end{bmatrix}, \begin{bmatrix} \mathbf{Q}^y \\ 0 \end{bmatrix} \right), \quad (1)$$

```

features_initial=Find_object_points(initial_image)
search_tree=Create_search_tree(features_initial)
while Images to treat do
    imagei=capture_image()
    features_frame=Find_points(imagei)
    matches=Search_features(search_tree, features_frame)
    [inliers_model, inliers_frame, num_inliers]=RANSAC(matches)
    if num_inliers>3 then
        [M]=compute_shape_matrix(inliers_initial)
        [S, Σ]=Kalman_filter(inliers_initial, M-1, inliers_frame)
        posei=from_shape_to_3d(S)
        covi=propagate_covariance_UT(S, Σ)
    end
end

```

Algorithm 1. Algorithm to use points instead of an active contour in pose computation.

where Q^x and Q^y are the x and y components respectively of the object points in the initial image. When the robot moves, a new image of the object is acquired, and the new point projections Q' are related to the initial ones Q through

$$Q' - Q = WS, \quad (2)$$

where

$$S = (t_x, t_y, M_{11} - 1, M_{22} - 1, M_{21}, M_{12}) , \quad (3)$$

is the 6-dimensional *shape vector* that encodes the image deformation from the first to the second view. From this shape vector the pose of the camera relative to the initial view can be computed [20] and also its covariance [1]. Algorithm 1 shows how to apply active contour pose estimation using point matches instead of the control points of the B-spline parameterisation of active contours².

2. Depth estimation from two images taken with different focal lengths

Recently, a depth estimation algorithm has been proposed when a zooming camera is available [4]. Using geometric calibration, the relative depth between the camera and an object can be recovered. With this algorithm there is no need of calibrating the camera intrinsic parameters.

Here we present an alternative approach to that based on a zooming camera. In order to simulate human foveated vision, a common strategy is to use two different cameras [23,22]: one with long focal length simulates the central visual region, with narrow field of view and high resolution, and another camera with a shorter focal length simulates the peripheral view, with wider field of view and poorer resolution. Other strategies to obtain foveated images include downsampling images [21] and using special

²Compare with algorithms presented in [1].

lenses [16]. An interesting alternative to camera-based foveal imaging has been proposed recently using a rotating laser scan [15].

Once the robot has oriented the head to see the object with the foveal camera, object projection is into the field of view of both cameras. This situation is equivalent to a zooming camera that takes two images of the same object using two different and known zoom positions.

One assumption was made in the depth-from-zoom algorithm: each zoom controller position should correspond always to the same focal length. In other words, the zoom controller is supposed to have good repetivity. With the presented setup (two cameras per eye), this assumption is no longer necessary, as focal length is fixed for both cameras and never changes.

A common simplification is to consider that the projection centers of both cameras are coincident [26], which is reasonable for some applications. Some zoom camera algorithms also make this assumption, even if it is well known that the optical center changes when zooming. The relation between foveal and peripheral images of a target object is only an affine scaling factor if centers are coincident, and target objects are centered in the projection center of each camera. As we cannot guarantee these facts we cannot restrict deformation only to scalings, and we should allow also for translations in the image plane directions. Thus, we use the reduced shape matrix [4]

$$\mathbf{W}_{\text{zoom}} = \left(\begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \begin{bmatrix} \mathbf{Q}^x \\ \mathbf{Q}^y \end{bmatrix} \right), \quad (4)$$

and compute the reduced shape vector

$$\mathbf{S} = [t_x, t_y, \rho]. \quad (5)$$

where ρ is the affine scaling factor.

A change in focal length causes a deformation of the object projection in the image that can be parameterised using a shape vector \mathbf{S} . If two scaling factors ρ_1 and ρ_2 are computed at two known depths z_1 and z_2 using the same difference in focal lengths, then the unknown depth z of an object can be recovered using the computed scaling factor ρ by applying [1]

$$\frac{z_2 - z_1}{z - z_1} = \frac{\rho(\rho_2 - \rho_1)}{\rho_2(\rho - \rho_1)}. \quad (6)$$

3. Experiments

3.1. Depth estimation in a foveal-peripheral camera configuration

We will use the active head of the Armar-III robot [6] from Karlsruhe University as our experimental platform (Figure 1). The head of Armar-III has two firewire 640×480 cameras per eye with different focal lengths, each one providing 25fps. The focal length of the foveal camera has been set to acquire focused images from distances larger than

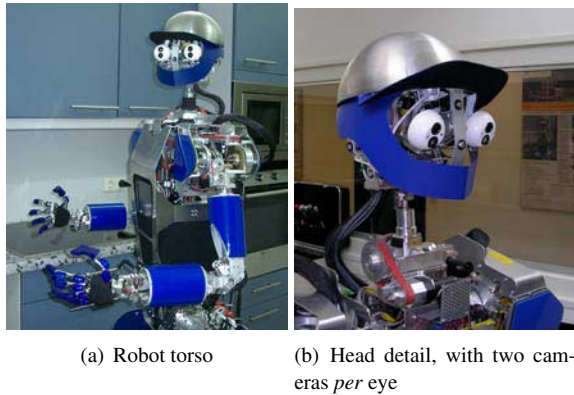


Figure 1. Armar III robot used in the experiments

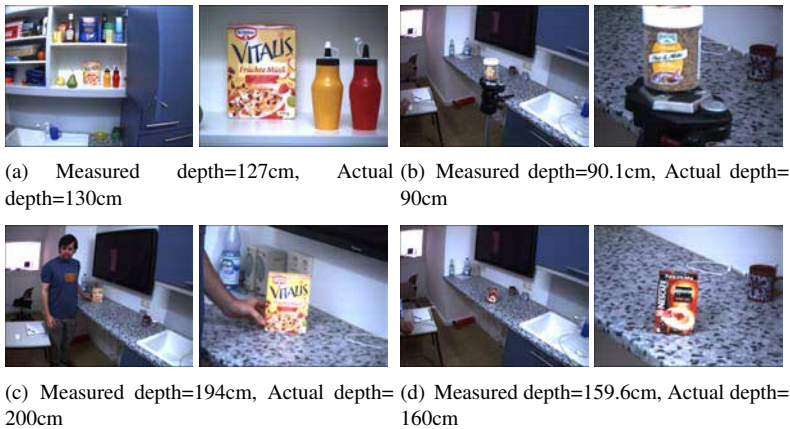


Figure 2. Peripheral and foveal image pairs (respectively) and the measured depth. Observe that image projection sizes are very similar in each camera for pair 2(a) and 2(b), and also for pair 2(c) and 2(d). This is obviously due to the fact that different object sizes are compensated by the different distances. However, our algorithm successfully recovers actual object depths.

100cm. Observe image defocusing in the foveal image (Figure 2(b)) when the distance from camera to object is only 90cm. We have experimented also with larger distances, but due to defocusing in the foveal images, point position is not precisely measured and, consequently, depth estimation sometimes fails.

Note that for these experiments only one eye, composed of two cameras, has been used. Cameras are not active, so the focal distance is fixed and the depth-of-field also. This is particularly restrictive for the foveal camera, as it has a large focal length. Other methods to obtain depth are difficult to apply here: depth-from-defocus is not applicable as we cannot control the focusing mechanism; depth-from-stereo is badly conditioned, as vergence between peripheral and foveal cameras is nearly 0, and the baseline is very short (approx. 4mm).

Results are depicted in Figure 2, with captions for each experiment. Observe that the sizes in the image of the objects of interest (a cereal box and a small spices jar) in Figures 2(a) and 2(b) are very similar. Actually, they are very different in size but

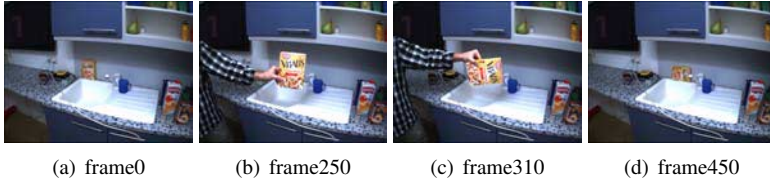


Figure 3. Peripheral images of the box sequence.



Figure 4. Foveal images of the box sequence.

one is closer to the camera than the other. This is the well known depth-size ambiguity in monocular imaging: with no additional information about the object or the scene, all translations can be recovered only up to a scale factor. However, our geometric calibration algorithm successfully recovers the depth, and consequently eliminates this ambiguity³. The same applies to Figures 2(c) and 2(d).

3.2. Metric motion estimation in monocular images

We apply Algorithm 1 presented before to a sequence of a textured object moved freely. Point matches are computed using the Sift algorithm. Figures 3 and 4 show some frames of the video sequence captured by the peripheral and the foveal cameras respectively. As it can be observed, the performed motion includes object translation and rotation.

Motion is estimated using foveal images, as they provide more resolution. Here the cameras are static, but in the future the idea is to control the head motion to maintain the object in the field of view of the foveal camera using the peripheral image.

Results of the motion estimation algorithm can be observed in Figure 5. As expected, variance (Figure 5(a)) is not the same for all pose components. T_z is recovered with higher variance than T_x and T_y , and R_z is recovered with lower variance than R_x and R_y . Observe that when the object approaches the camera, in the middle of the sequence, the variance of T_z diminishes. R_z pose component is always recovered properly despite the change in orientation the object suffered in the middle of the sequence. The path traveled by the object (Figure 5(b)) can be reconstructed metrically using a camera calibration method to recover the focal length of the foveal camera, and with the initial distance recovered with the algorithm presented in Section 2.

³Note that in order to obtain metric T_x and T_y translations the focal length should be also known, but it is not necessary for T_z computation.

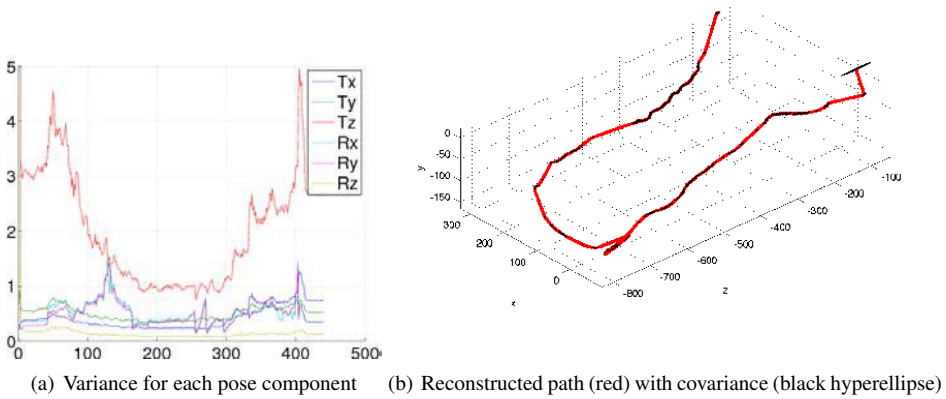


Figure 5. Metric motion estimation for the box sequence.

4. Conclusions

We have shown that it is possible to apply active contour algorithms to recover object motion considering point matches instead of control points in a B-spline parameterisation. Active contours are interesting because they are easy to track and can be used in poorly textured objects. Contrarily, point correspondences are easy to find in richly textured objects. We believe that active contours and point matches complement in a very natural way.

We have presented a depth-from-zooming algorithm that assumes repeatability of the zoom controller: the same zoom controller position corresponds exactly to the same equivalent focal length in the pinhole camera model. Also, the projection rays of the camera when performing the zoom should change in order to observe changes useful for depth estimation. We have shown here that these assumptions can be removed when a foveal-peripheral camera configuration is used, with the advantage that the projection centers are physically not the same.

Acknowledgements

The authors would like to thank Prof. Rüdiger Dillmann and Dr. Tamim Asfour for inviting us to their institute and for the help offered to carry out experiments on the Armar-III robot.

References

- [1] G. Alenyà. *Estimació del moviment de robots mitjançant contorns actius*. PhD thesis, Universitat Politècnica de Catalunya, 2007.
- [2] G. Alenyà, J. Escoda, A.B. Martínez, and C. Torras. Using laser and vision to locate a robot in an industrial environment: A practical experience. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3539–3544, Barcelona, April 2005.

- [3] G. Alenyà, E. Martínez, and C. Torras. Fusing visual and inertial sensing to recover robot egomotion. *Journal of Robotic Systems*, 21(1):23–32, 2004.
- [4] G. Alenyà and C. Torras. Depth from the visual motion of a planar target induced by zooming. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 4727–4732, 2007.
- [5] G. Alenyà and C. Torras. Zoom control to compensate camera translation within a robot egomotion estimation approach. In *Sixth International Workshop on Robot Motion and Control*, volume 360 of *Lecture Notes in Control and Information Sciences*, pages 81–88, 2007.
- [6] T. Asfour, K. Regenstein, P. Azad, J. Schröder, A. Bierbaum, N. Vahrenkamp, and R. Dillmann. Armar-III: An integrated humanoid platform for sensory-motor control. In *Proceedings of the 6th IEEE-RAS International Conference on Humanoid Robots*, Genoa, December 2006.
- [7] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In *Proceedings of the 9th European Conference on Computer Vision*, volume 3951 of *Lecture Notes in Computer Science*, pages 404–417, Graz, 2006. Springer-Verlag.
- [8] P.A. Beardsley, A. Zisserman, and D.W. Murray. Sequential updating of projective and affine structure from motion. *International Journal of Computer Vision*, 23(3):235–259, 1997.
- [9] A. Blake and M. Isard. *Active contours*. Springer, 1998.
- [10] T. Cham and R. Cipolla. Automated b-spline curve representation incorporating mdl and error-minimizing control point insertion strategies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(1), 1999.
- [11] T. Drummond and R. Cipolla. Application of Lie algebras to visual servoing. *International Journal of Computer Vision*, 37(1):21–41, 2000.
- [12] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2 edition, 2004.
- [13] R. Horaud, F. Dornaika, B. Lamiroy, and S. Christy. Object pose: the link between weak perspective, paraperspective, and full perspective. *International Journal of Computer Vision*, 22(2):173–189, 1997.
- [14] J. Koenderink and A. J. van Doorn. Affine structure from motion. *J. Opt. Soc. Am. A*, 8(2):377–385, 1991.
- [15] P. Kohlhepp, G. Bretthauer, M. Walther, and R. Dillmann. Using orthogonal surface directions for autonomous 3d-exploration of indoor environments. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3086–3092, Beijing, October 2006.
- [16] K. Kuniyoshi, N. Kita, K. Sugimoto, S. Nakamura, and T. Suehiro. A foveated wide angle lens for active vision. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2982–2988, Nagoya, May 1995.
- [17] V. Lepetit and P. Fua. Keypoint recognition using randomized trees. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(9):1465–1479, 2006.
- [18] Y. Liu, T.S. Huang, and O.D. Faugeras. Determination of camera location from 2d to 3d lines and point correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):28–37, 1990.
- [19] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [20] E. Martínez and C. Torras. Qualitative vision for the guidance of legged robots in unstructured environments. *Pattern Recognition*, 34(8):1585–1599, 2001.
- [21] I.D. Reid and D.W. Murray. Active tracking of foveated feature clusters using affine structure. *International Journal of Computer Vision*, 18(1):41–60, April 1996.
- [22] B. Scassellati. A binocular, foveated active vision system. Technical Report 1628, MIT, Artificial Intelligence Laboratory, 1999.
- [23] J. Schiehlen and E.D. Dickmanns. Design and control of a camera platform for machine vision. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2058–2063, 1994.
- [24] L. S. Shapiro, A. Zisserman, and M. Brady. 3D motion recovery via affine epipolar geometry. *International Journal of Computer Vision*, 16(2):147–182, 1995.
- [25] B. Tordoff and D. Murray. Reactive control of zoom while fixating using perspective and affine cameras. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1):98–112, January 2004.
- [26] A. Ude, C. Gaskett, and G. Cheng. Foveated vision systems with two cameras per eye. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3457–3462, May 2006.
- [27] H. Yang, W. Wang, and J. Sun. Control point adjustment for b-spline curve approximation. *Computer-Aided Design*, 36(7):639–652, 2004.

The SLAM problem: a survey

Josep AULINAS^a Yvan PETILLOT^b Joaquim SALVI^a and Xavier LLADÓ^a

^a*Institute of Informatics and Applications, University of Girona, Girona (Spain)*

^b*Ocean Systems Lab, Heriot-Watt University, Edinburgh (UK)*

Abstract. This paper surveys the most recent published techniques in the field of Simultaneous Localization and Mapping (SLAM). In particular it is focused on the existing techniques available to speed up the process, with the purpose to handel large scale scenarios. The main research field we plan to investigate is the filtering algorithms as a way of reducing the amount of data. It seems that almost all the current approaches can not perform consistent maps for large areas, mainly due to the increase of the computational cost and due to the uncertainties that become prohibitive when the scenario becomes larger.

Keywords. SLAM, Kalman filter, Particle Filter, Expectation Maximization

Introduction

Simultaneous Localization and Mapping (SLAM) also known as Concurrent Mapping and Localization (CML) is one of the fundamental challenges of robotics, dealing with the necessity of building a map of the environment while simultaneously determining the location of the robot within this map. The aim of this paper is to survey the advantages and disadvantages of current available techniques, to compare and contrast them, and finally to identify gaps, i.e. possible new research directions or further improvements. This survey is conducted as the starting point of a bigger project, which involves computer vision in SLAM (Visual SLAM) in underwater scenarios.

SLAM is a process by which a mobile robot can build a map of an environment and at the same time use this map to deduce its location. Initially, both the map and the vehicle position are not known, the vehicle has a known kinematic model and it is moving through the unknown environment, which is populated with artificial or natural landmarks. A simultaneous estimate of both robot and landmark locations is required. The SLAM problem involves finding appropriate representation for both the observation and the motion models [1]. In order to do so, the vehicle must be equipped with a sensorial system capable of taking measurements of the relative location between landmarks and the vehicle itself.

Several research groups and researchers have worked and are currently working in SLAM, and the most commonly used sensors can be categorized into laser-based, sonar-based, and vision-based systems. Additional sensorial sources are used to better perceive robot state information and the outside world [2], such as, compasses, infrared technology and Global Positioning System (GPS). However, all these sensors carry certain errors, often referred to as measurement noise, and also have several range limitations

making necessary to navigate through the environment, for instance, light and sound cannot penetrate walls.

Laser ranging systems are accurate active sensors. Its most common form operates on the time of flight principle by sending a laser pulse in a narrow beam towards the object and measuring the time taken by the pulse to be reflected off the target and returned to the sender. *Sonar-based systems* are fast and provide measurements and recognition capacities with amounts of information similar to vision, but with the lack of appearance data. However, its dependence on inertial sensors, such as odometers, implies that a small error can have large effects on later position estimates [2]. On the other hand, *Vision systems* are passive, they have long range and high resolution, but the computation cost is considerably high and good visual features are more difficult to extract and match. Vision is used to estimate the 3D structure, feature location and robot pose, for instance by means of stereo camera pairs or monocular cameras with structure from motion recovery.

Further classification can be made in terms of working environment, for instance, ground indoor, ground outdoor, air-borne or underwater. Most of the work done so far focuses on ground and mainly indoor environments [3] [4] [5] [6], only few papers deal with airborne applications [7] [8] and a few more present the SLAM in underwater conditions and they generally work with acoustic data [9]. Recently, there is a growing interest in SLAM for underwater scenarios [10], in which vision plays an important role [11] [12] [13], in most cases combined with other sensory systems to acquire both depth and appearance information of the scene, for instance, acoustic or inertial sensors.

The representation of the observation and the motion models is generally performed by computing its prior and posterior distributions using probabilistic algorithms, which are briefly described in section 1. These algorithms are strongly influenced by the data measurement and association, which are presented in section 2. Existing literature is classified in section 3, listing the main advantages and disadvantages of each group. Finally, section 4 summarizes the main ideas and an overall discussion is given.

1. Solutions to the SLAM Problem : Filters in SLAM

Robotic map-building can be traced back to 25 years ago, and since the 1990s probabilistic approaches (i.e. Kalman Filters (KF), Particle Filters (PF) and Expectation Maximization (EM)) have become dominant in SLAM. The three techniques are mathematical derivations of the recursive Bayes rule. The main reason for this probabilistic techniques popularity is the fact that robot mapping is characterized by uncertainty and sensor noise, and probabilistic algorithms tackle the problem by explicitly modeling different sources of noise and their effects on the measurements [2].

1.1. Kalman Filters and its variations (KF)

Kalman filters are Bayes filters that represent posteriors using Gaussians, i.e. unimodal multivariate distributions that can be represented compactly by a small number of parameters. KF SLAM relies on the assumption that the state transition and the measurement functions are linear with added Gaussian noise, and the initial posteriors are also Gaussian. There are two main variations of KF in the state-of-the-art SLAM: the Extended Kalman Filter (EKF) and its related Information Filtering (IF) or Extended IF (EIF). The

EKF accommodates the nonlinearities from the real world, by approximating the robot motion model using linear functions. Several existing SLAM approaches use the EKF [3] [5] [14] [15]. The IF is implemented by propagating the inverse of the state error covariance matrix. There are several advantages of the IF filter over the KF. Firstly, the data is filtered by simply summing the information matrices and vector, providing more accurate estimates [16]. Secondly, IF are more stable than KF [17]. Finally, EKF is relatively slow when estimating high dimensional maps, because every single vehicle measurement generally effects all parameters of the Gaussian, therefore the updates requires prohibitive times when dealing with environments with many landmarks [18].

However, IF have some important limitations, a primary disadvantage is the need to recover a state estimate in the update step, when applied to nonlinear systems. This step requires the inversion of the information matrix. Further matrix inversions are required for the prediction step of the information filters. For high dimensional state spaces the need to compute all these inversions is generally believed to make the IF computationally poorer than the Kalman filter. In fact, this is one of the reasons why the EKF has been vastly more popular than the EIF [19]. These limitations do not necessarily apply to problems in which the information matrix possesses structure. In many robotics problems, the interaction of state variables is local; as a result, the information matrix may be sparse. Such sparseness does not translate to sparseness of the covariance. Information filters can be thought of as graphs, where states are connected whenever the corresponding off-diagonal element in the information matrix is non-zero. Sparse information matrices correspond to sparse graphs. Some algorithms exist to perform the basic update and estimation equations efficiently for such fields [20] [21], in which the information matrix is (approximately) sparse, and allow to develop an extended information filter that is significantly more efficient than both Kalman filters and non sparse Information Filter.

The Unscented Kalman Filter (UKF) [22] addresses the approximation issues of the EKF and the linearity assumptions of the KF. KF performs properly in the linear cases, and is accepted as an efficient method for analytically propagating a Gaussian Random Variable (GRV) through a linear system dynamics. For non linear models, the EKF approximates the optimal terms by linearizing the dynamic equations. The EKF can be viewed as a first-order approximation to the optimal solution. In these approximations the state distribution is approximated by a GRV, which then is propagated analytically through the first-order linearization of the nonlinear system. These approximations can introduce large errors in the true posterior mean and covariance, which may lead sometimes to divergence of the filter. In the UKF the state distribution is again represented by a GRV, but is now specified using a minimal set of carefully chosen sample points. These sample points completely capture the true mean and covariance of the GRV, and when propagated through the true non-linear system, captures the posterior mean and covariance accurately to the 3rd order for any nonlinearity. In order to do that, the unscented transform is used.

One of the main drawbacks of the EKF and the KF implementation is the fact that for long duration missions, the number of landmarks will increase and, eventually, computer resources will not be sufficient to update the map in real-time. This scaling problem arises because each landmark is correlated to all other landmarks. The correlation appears since the observation of a new landmark is obtained with a sensor mounted on the mobile robot and thus the landmark location error will be correlated with the error in the vehicle location and the errors in other landmarks of the map. This correlation is of

fundamental importance for the long-term convergence of the algorithm, and needs to be maintained for the full duration of the mission. The Compressed Extended Kalman Filter (CEKF) [23] algorithm significantly reduces the computational requirement without introducing any penalties in the accuracy of the results. A CEKF stores and maintains all the information gathered in a local area with a cost proportional to the square of the number of landmarks in the area. This information is then transferred to the rest of the global map with a cost that is similar to full SLAM but in only one iteration.

The advantage of KF and its variants is that provides optimal Minimum mean-square Error (MMSE) estimates of the state (robot and landmark positions), and its covariance matrix seems to converge strongly. However, the Gaussian noise assumption restricts the adaptability of the KF for data association and number of landmarks.

1.2. Particle Filter based methods (PF)

PF, also called the sequential Monte-Carlo (SMC) method, is a recursive Bayesian filter that is implemented in Monte Carlo simulations. It executes SMC estimation by a set of random point clusters ('particles') representing the Bayesian posterior. In contrast to parametric filters (e.g., KF), PF represents the distribution by a set of samples drawn from this distribution, what makes it capable of handling highly nonlinear sensors and non-Gaussian noise. However, this ability produces a growth in computational complexity on the state dimension as new landmarks are detected, becoming not suitable for real time applications [24]. For this reason, PF has only been successfully applied to localization, i.e. determining position and orientation of the robot, but not to map-building, i.e. landmark position and orientation; therefore, there are no important papers using PF for the whole SLAM framework, but there exist few works that deal with the SLAM problem using a combination of PF with other techniques, for instance, the FastSLAM [24] and the fastSLAM2.0 [25]. FastSLAM takes advantage of an important characteristic of the SLAM problem (with known data association): landmark estimates are conditionally independent given the robot's path [26]. FastSLAM algorithm decomposes the SLAM problem into a robot localization problem, and a collection of landmark estimation problems that are conditioned on the robot pose estimate. A key characteristic of FastSLAM is that each particle makes its own local data association. In contrast, EKF techniques must commit to a single data association hypothesis for the entire filter. In addition FastSLAM uses a particle filter to sample over robot paths, which requires less memory usage and computational time than a standard EKF or KF.

1.3. Expectation Maximization based methods (EM)

EM estimation is a statistical algorithm that was developed in the context of maximum likelihood (ML) estimation and it offers an optimal solution, being an ideal option for map-building, but not for localization. The EM algorithm is able to build a map when the robot's pose is known, for instance, by means of expectation [27]. EM iterates two steps: an expectation step (E-step), where the posterior over robot poses is calculated for a given map, and maximization step (M-step), in which the most likely map is calculated given these pose expectations. The final result is a series of increasingly accurate maps. The main advantage of EM with respect to KF is that it can tackle the correspondence problem (data association problem) surprisingly well [2]. This is possible thanks to

the fact that it localizes repeatedly the robot relative to the present map in the E-step, generating various hypotheses as to where the robot might have been (different possible correspondences). In the latter M-step, these correspondences are translated into features in the map, which then get reinforced in the next E-step or gradually disappear. However, the need to process the same data several times to obtain the most likely map makes it inefficient, not incremental and not suitable for real-time applications [28]. Even using discrete approximations, when estimating the robot's pose, the cost grows exponentially with the size of the map, and the error is not bounded; hence the resulting map becomes unstable after long cycles. These problems could be avoided if the data association was known [29], what is the same, if the E-step was simplified or eliminated. For this reason, EM usually is combined with PF, which represents the posteriors by a set of particles (samples) that represent a guess of the pose where the robot might be. For instance, some practical applications use EM to construct the map (only the M-step), while the localization is done by different means, i.e. using PF-based localizer to estimate poses from odometer readings [2].

2. Measuring and Data Association

The most fundamental key topic into all SLAM solutions is the *data association* problem, which arises when landmarks cannot be uniquely identified, and due to this the number of possible hypotheses may grow exponentially, making absolutely unviable the SLAM for large areas. Data association in SLAM can be simply presented as a feature correspondence problem, which identifies two features observed in different positions and different points in time as being from the same physical object in the world. Two common applications of such data association are to match two successive scenes and to close a loop of a long trajectory when a robot comes to the starting point of the trajectory again.

So far most computer vision approaches only uses 2D information to perform data association, but in underwater scenarios this data association is more complicated due to more significant levels of distortion and noise. Therefore, in order to succeed when solving the correspondence problem very robust features are necessary, even under weak lighting conditions or under different points of view. The use of vision sensors offers the possibility to extract landmarks considering 2D and 3D information [11] [30], hence more robust features can be selected.

Feature recognition, tracking and 3D reconstruction are important steps that feed the measurements to the SLAM framework. Feature tracking is the problem of estimating the locations of features in an image sequence (for instance, Harris corner detector and Random Sample Consensus (RANSAC), Scale Invariant Feature Transform (SIFT) [15] or Speeded-up Robust Features (SURF) [31]). 3D reconstruction is the problem of obtaining the 3D coordinates and the camera pose using two or more 2D images (for instance, by using epipolar geometry and fundamental matrix). Fortunately, recent advances in computer vision techniques and, more precisely, in feature extraction enable the usage of high-level vision-based landmarks (complex and natural structures) in contrast to early attempts using low-level features (e.g., vertical edges, line segments, etc.) and artificial beacons. However, the use of vision has several limitations and practical difficulties, for example, several assumptions about the environment must be done in order to simplify

the problem (similarly to what humans do using prior knowledge to make decisions on what their eyes see); in regions without texture or with repeating structures there is no method of finding true matches; and even if this matching problem is solved, images are always noisy.

The *loop-closing* problem (a robot turns back to the starting point of its trajectory) requires successful identification of revisited landmarks to build a consistent map in large scale environments. Due to accumulated errors along the trajectory (drift), the reconstructed map is not consistent, i.e., the loop of the trajectory is not closed properly. Correct data association is required to uniquely identify the landmarks corresponding to previously seen ones, from which loop-closing can be detected. Then, different techniques are applied to correct the map, for example, Kalman smoother-based (used by most of the current solutions to the SLAM problem) and EM-based techniques [27].

3. Classification: Pros and Cons

From the previous section, it seems clear that few works have been published on underwater SLAM [5] [13] and even less on underwater visual SLAM [30]. Most of the underwater approaches use sonar or other non visual sensory systems. There exist various V-SLAM approaches for terrestrial applications [3] [4] [14] [15], most of them deal with the uncertainties by using Kalman Filters (KF) and its variation Extended Kalman Filters (EKF) [3] [14] [15], and another group of papers uses some improved information filter [16] [17] [20] [21], i.e. sparse expanded information filter (SEIF).

It seems obvious that almost none of the current approaches can perform consistent maps for large areas, mainly due to the increase on computational cost and on the uncertainties. Therefore this is possibly the most important issue that needs to be improved. Some recent publications tackle the problem by using multiple maps, or sub-maps that are lately used to build a larger global map [4] [32] [33]. However these methods rely considerably on assuming proper data association, which is another important issue that needs to be improved. Table 1 provides a list of advantages and disadvantages of different SLAM strategies in terms of the method used to deal with uncertainties.

Essentially, the most challenging methods not still solved are the ones enabling large-scale implementations in increasingly unstructured environments, i.e. underwater, and especially in situations where other current solutions are unavailable or unreliable. According to the bibliographical survey, SLAM solutions could be improved either by formulating more efficient and consistent to large scenarios filtering algorithms, and solving in a very robust way the data association problem. For the first case, different filters applied into the SLAM framework must be studied, for instance the compressed extended Kalman Filter (CEKF), the Unscented Kalman Filter (UKF) or the information filters (IF/EIF). The second issue is currently solved using SIFT and SURF, which seem to be considerably good solutions for the data association problem, however they become computationally expensive when dealing with high dimensional maps.

4. Discussion

This survey allows to find the most interesting filtering techniques and identify many of its particularities. These filtering strategies are Kalman Filter (KF), Information Filter

Table 1. List of advantages and disadvantages of filtering approaches applied into the SLAM framework.

| <i>Pros</i> | <i>Cons</i> |
|---|--|
| Kalman Filter and Extended KF (KF/EKF) [3] [5] [14] [15] | |
| - high convergence | - Gaussian assumption |
| - handle uncertainty | - slow in high dimensional maps |
| Compressed Extended KF (CEKF) [23] | |
| - reduced uncertainty | - require very robust features |
| - reduction of memory usage | - data association problem |
| - handle large areas | - require multiple map merging |
| - increase map consistency | |
| Information Filters (IF) [16] [17] | |
| - stable and simple | - data association problem |
| - accurate | - may need to recover a state estimates |
| - fast for high dimensional maps | - in high-D is computationally expensive |
| Particle Filter (PF) [24] [25] | |
| - handle nonlinearities | - growth in complexity |
| - handle non-Gaussian noise | |
| Expectation Maximization (EM) [16] [27] | |
| - optimal to map building | - inefficient, cost growth |
| - solve data association | - unstable for large scenarios |
| | - only successful in map building |

(IF), Unscented Kalman Filter (UKF) and Compressed Kalman Filter (CKF). A general classification of the current filtering strategies is given, contrasting the pros and cons.

The most interesting outcome from the survey is that for large scenarios, or maps with high population of landmarks, the CKF seems to be better as compared to other methods. When dealing with these kind of maps, the state vector and its associated covariance matrix keeps growing with the quantity of landmarks observed. This growth makes the mathematical operations more complex and increases dramatically the time consumption, i.e. the computational cost. The strategy used by the CKF to compute local KFs and then update its output to a global map seems really consistent, because it only needs to handle with small amounts of data during the local iteration process.

As always, there is room for additional improvement. Further areas of research that have been identified encouraging future investigation are:

1. *Test and adapt all methods to non-linear problems.* this means to implement and test linear models, and then improve them by implementing its extended formulations, for instance, Extended Kalman Filter (EKF), Extended Information Filter(EIF) and Compressed Extended Kalman Filter (CEKF).
2. *Find new solutions for non-gaussian noise problems.* Although gaussian noise is assumed in all models presented so far, not always reflects the problems of the real world. It seems that UKF could handle with different types of noise, but this topic has not been investigated in deep yet.
3. *Develop Visual SLAM algorithms useful for navigation purposes on underwater vehicles and for 3D reconstruction of the seafloor.* In fact, this is the ultimate destination of this work, which should contribute to improve current underwater

vehicles and underwater research. This survey is a first step to this whole work, bringing insightful information on filtering techniques. As far as data association problem is concerned, it is well known that it is completely dependent on the ability to find proper features, which in underwater scenarios is a considerably critical task, mainly due to the enormous amount of noise, the dynamism of the environment and the lack of significant landmarks. Obviously if this landmark detection is not robust enough there won't be possibility for the SLAM to work. For this reason, the use of vision to define landmarks seems to open a wide range of different possibilities. In addition a vision system is considerably less expensive than other sensorial systems, for instance, sonar. The main drawback lay in its high computational cost to extract and match features. Further problems that needs to be addressed from the limitations that arises from using computer vision in SLAM are the dynamism of the robot environment, i.e. changes of location of other agents in the environment, creates a big challenge, because it introduces inconsistent sensor measurements and because there are almost no algorithms that can dealt with mapping in dynamic environments.

References

- [1] H. Durrant-White and T. Bailey. Simultaneous localization and mapping. *IEEE Robotics and Automation magazine*, 13(2):99–108, 2006.
- [2] S. Thrun. Robotic mapping: A survey. *Exploring Artificial Intelligence in the New Millenium*. The Morgan Kaufmann Series in Artificial Intelligence (Hardcover) by Gerhard Lakemeyer (Editor), Bernhard Nebel (Editor). ISBN ISBN-10: 1558608117, 2002
- [3] A.J. Davison and D. Murray. Simultaneous localization and map-building using active vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):865–880, 2002.
- [4] C. Estrada, J. Neira and J.D. Tardós. Hierarchical SLAM: Real-time accurate mapping of large environments. (*IEEE*) *Transactions On Robotics*, 21(4):588–596, 2005.
- [5] P.M. Newman and J.J. Leonard. Consistent, convergent, and constant-time SLAM. *International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco, Mexico, pp. 1143–1150, 2003.
- [6] P.M. Newman. On the structure and solution of the simultaneous localization and mapping problem. *PhD Thesis, University of Sydney*, 1999.
- [7] J. Kim and S. Sukkariéh. Airborne simultaneous localisation and map building. *Proceedings - IEEE International Conference on Robotics and Automation*, Vol. 1, pp. 406–411, 2003.
- [8] J. Kim and S. Sukkariéh. Autonomous airborne navigation in unknown terrain environments. *IEEE Transactions on Aerospace and Electroninc Systems*, 40(3):1031–1045, 2004.
- [9] S.B. Williams, P. Newman, M.W.M.G. Dissanayake, and H.F. Durrant-Whyte. Autonomous underwater simultaneous localisation and map building. *Proceedings of IEEE International Conference on Robotics and Automation*, San Francisco, USA, pp. 1143–1150, 2000.
- [10] R.M. Eustice, H. Singh, J. Leonard and M. Walter. Visually mapping the RMS titanic: Conservative covariance estimates for SLAM information filters. *IEEE Transactions on Robotics*, Vol. 22(6), pp. 1100–1114, 2006.
- [11] J.M. Sáez, A. Hogue, F. Escolano and M. Jenkin. Underwater 3D SLAM through entropy minimization. *Proceedings - IEEE International Conference on Robotics and Automation 2006*, art. no. 1642246, pp. 3562–3567, 2006.
- [12] O. Pizarro, R. Eustice and H. Singh. Large area 3D reconstructions from underwater surveys. *Ocean '04 - MTS/IEEE Techno-Ocean: Bridges across the Oceans - Conference Proceedings*, Vol. 2, pp. 678–687, 2004.
- [13] R.M. Eustice. Large-area visually augmented navigation for autonomous underwater vehicles. *PhD Thesis*, 2005.

- [14] P. Jensfelt, D. Kragic, J. Folkesson and M. Björkman. A framework for vision based bearing only 3D SLAM. *Proceedings - IEEE International Conference on Robotics and Automation, ICRA*, art. no. 1641990, pp. 1944–1950, 2006.
- [15] S. Se, D. Lowe and J. Little. Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. *The international Journal of robotics Research*, 21(8):735–758, 2002.
- [16] S. Thrun and Y. Liu. Multi-robot SLAM with sparse extended information filters. *Proceedings of the 11th International Symposium of Robotics Research (ISRR'03)*, Sienna, Italy, 2003. Springer.
- [17] S. Thrun, C. Martin, Y. Liu, D. Hähnel, R. Emery-Montemerlo, D. Chakrabarti, and W. Burgard. A real-time expectation maximization algorithm for acquiring multi-planar maps of indoor environments with mobile robots. *IEEE Transactions on Robotics and Automation*, 20(3):433–442, 2004.
- [18] S. Thrun, Y. Liu, D. Koller, A.Y. Ng, Z. Ghahramani and H. Durrant-Whyte. Simultaneous localization and mapping with sparse extended information filters. *International Journal of Robotics Research*, 23(7-8):693–716, 2004.
- [19] S. Thrun, D. Burgard and W. Fox. Probabilistic Robotics. *MIT Press*, 2005. ISBN-10:0-262-20162-3
- [20] M.R. Walter, R.M. Eustice and J.J. Leonard. A provably consistent method for imposing exact sparsity in feature-based SLAM information filters. *Proceedings of the 12th International Symposium of Robotics Research (ISRR)*. pp. 241–234, 2007.
- [21] M.R. Walter, R.M. Eustice and J.J. Leonard. Exactly sparse extended information filters for feature based SLAM. *The International Journal of Robotics Research*, 26(4):335–359, 2007.
- [22] E. Wan and R. van der Merwe. Kalman Filtering and Neural Networks, Chapter 7: The Unscented Kalman Filter. *Wiley*, 2001. ISBN: 978-0-471-36998-1
- [23] J.E. Guivant and E.M. Nebot. Optimization of the Simultaneous Localization and Map-Building Algorithm for Real-Time Implementation. *IEEE Transactions on Robotics and Automation*, 17(3), 2001.
- [24] M. Montemerlo, S. Thrun, D. Koller and B. Wegbreit. FastSLAM: A factored solution to the simultaneous localization and mapping problem. *Proceedings of the National Conference on Artificial Intelligence*, pp. 593–598, 2002.
- [25] M. Montemerlo, S. Thrun, D. Koller and B. Wegbreit. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges. *18th International Joint Conference on Artificial Intelligence (IJCAI)*, Acapulco Mexico, pp. 1151–1156, 2003.
- [26] M. Montemerlo, S. Thrun. FastSLAM: A Scalable Method for the simultaneous localization and mapping problem in robotics. *Springer Tracts in Advanced Robotics*, vol. 27, ISBN: 978-3-540-46399-3, 2007
- [27] W. Burgard, D. Fox, H. Jans, C. Matenar, and S. Thrun. Sonar-based mapping with mobile robots using EM. *Proceedings - 16th International Conference on Machine Learning*, 1999.
- [28] Z. Chen, J. Samarabandu and R. Rodrigo. Recent advances in simultaneous localization and map-building using computer vision. *Advanced Robotics*, 21(3-4):233–265, 2007.
- [29] S. Thrun. A probabilistic online mapping algorithm for teams of mobile robots. *International Journal of Robotics Research*, 20(5):335–363, 2001.
- [30] Y. Petillot, J. Salvi, B. Batlle. 3D Large-Scale Seabed Reconstruction for UUV Simultaneous Localization and Mapping. *IFAC Workshop on Navigation, Guidance and Control of Underwater Vehicles, NGCUV'08*, April 2008.
- [31] A.C. Murillo, J.J. Guerrero and C. Sagues. Surf features for efficient robot localization with omnidirectional images. *Proceedings - IEEE International Conference on Robotics and Automation*, art. no. 4209695, pp. 3901–3907, 2007.
- [32] S. Se, D. Lowe and J. Little. Vision-based global localization and mapping for mobile robots. *IEEE Transactions on Robotics*, 21(3):364–375, 2005.
- [33] L.M. Paz, P. Jensfelt, J.D. Tardós and J. Neira. EKF SLAM updates in $O(n)$ with divide and conquer SLAM. *IEEE International Conference on Robotics and Automation*, 4209325:1657–1663, 2007.

An approach for Mail-Robot navigation using a CBR technique

MARTÍ NAVARRO, STELLA HERAS and VICENTE JULIÁN

Universidad Politécnica de Valencia

Camí de Vera s/n

46220 Valencia, Spain

e-mail: mnavarro@dsic.upv.es, sheras@dsic.upv.es, vinglada@dsic.upv.es

Abstract. Multi-Agent System (MAS) paradigm has seemed especially appropriate for developing applications in complex domains, as Real-Time environments are. In this situation, an artificial intelligent technique may require tasks which involve searching in large search spaces in a bounded time. However, the main problem of MAS applications over real-time environments is to merge intelligent deliberative techniques with real-time actions in complex and distributed environments. In this settings, agents must make sure that they will be able to perform a service within a specific time before committing themselves to its performance. This paper proposes the use of a Case-Based Reasoning (CBR) technique to cope with this functionality. The technique is tested in an example where the tasks are bounded and need a temporal control.

Keywords. Real-Time, Multi-Agent Systems, Case-Based Reasoning.

1. Introduction

Complex deliberative processes, which allow agents to adapt and learn, are unbounded and to integrate them in real-time systems is difficult. Typically, in the multi-agent area the processes are carried out by deliberative agents, which decide what to do and how to do it according to their mental attitudes. In a deliberative agent, it is relatively simple to identify decision processes and how to perform them. However, its main drawback lies in finding a mechanism that allows its efficient and temporal bounded execution. Therefore, it would be interesting to integrate complex deliberative processes for decision-making in real-time systems in a simple and efficient way. This work presents an application example which incorporates a bounded deliberative case-based technique, which facilitates a temporal analysis for complex tasks in an agent with real-time restrictions. The aim of this example is to test the suitability of CBR to predict if agents will be able to perform a service on time before committing themselves to its performance.

The application consists in solving the automation of the internal mail management of a department that is physically distributed in a single floor of a build-

ing plant (a restricted and well-known test environment). The mail robot in charge of attending the mail requests is controlled by an agent specifically designed with an hybrid agent architecture with both real-time and deliberative capabilities. The rest of the paper is structured as follows: in section 2 the problem is presented; in section 3 the Robot Agent architecture is explained and finally, results and conclusions are shown in section 4.

2. Postman Service Problem

The problem to solve consists in the automated management of the internal and external mail (physical mail, non-electronic) in a department plant. The system created by this automation must be able to request the shipment of a letter or package from an office on one floor to another office on the same floor, as well as the reception of external mail at a collection point for later distribution. Once this service has been requested, a set of mobile robots must gather the shipment and direct it to the destination. It is important to note that each mail or package distribution must be ended before a maximum time, specified in the shipment request. This example is a complex problem and is clearly distributed, which makes the multi-agent system paradigm suitable for its resolution.

In the proposed example we can distinguish three types of agents: (i) **Interface Agent**: this agent is in charge of gathering user requests. A request will be transmitted by the Interface Agent to the mail service offered in the department plant (this service is offered by the Floor Agent). The user can employ a mobile device to communicate with an Interface Agent. (ii) **Floor Agent**: the mission of this agent is to gather the delivery/reception of mail sent by the Interface Agent and to distribute the work among the available Robot Agents around the plant. The Floor Agent interacts with the Robot Agents by means of the invocation of their mail delivery service. (iii) **Robot Agent**: the Robot Agent is in charge of controlling a physical robot and managing the mail list that this robot should deliver. In the proposed example, three Robot Agents are employed. Each Robot Agent controls a *Pioneer 2* mobile Robot. The Robot Agent must satisfy critical time restrictions, the tasks that control the robot sensors and effectors have temporal constraints. Moreover, this agent periodically sends information about its situation and state to the Floor Agent. This information is used by the Floor Agent to decide the most appropriate agent to send a new delivery/reception request. Next section shows the Robot Agent architecture in detail.

3. Robot Agent Architecture

Figure 1 shows the internal architecture of the Robot Agent, which is mainly composed of three modules:

- **Communication module**: this module is in charge of coding/decoding and sending/receiving the messages. The message is coded at this level. This codification converts the message into a useful load for the transport message. Additional and necessary information is added in order to support the platform communication model.

- **Navigation module:** this module guides the robot around the physical environment toward the next objective. The navigation through the environment is carried out using the information obtained by the sensors and the execution of some specific time-bounded tasks which are necessary to achieve a specific goal.
- **Temporal Constraint Analysis (TCA) module:** by means of this module, the agent can decide if it could perform a specific service before a deadline and hence, to commit itself to the execution of that service.

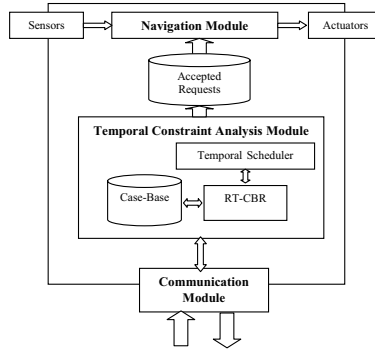


Figure 1. Internal architecture of the Robot Agent.

The TCA module must decide if an agent has enough time to perform a specific service. A possible way of performing such decision-making functionality is to use the knowledge that the manager has gained about previous analysis that it undertook in the past. This fits the main assumption of CBR systems, which adapt previous problem solving cases to cope with current similar problems [1]. Therefore, in the absence of unpredictable circumstances, we assume that an agent has enough time to perform a service if it has already succeeded in doing so in a similar situation.

To carry out the decision-making about contracting or not a commitment to perform the service, the TCA module has been enhanced with a RT-CBR module, following a soft Real-Time approach. This module estimates the time that the performance of a service could entail by using the time spent in performing similar services. With this aim, the RT-CBR module follows the typical steps of the CBR methodology: *retrieve* similar experiences from a *case-base*, *reuse* the knowledge acquired in them, *revise* such knowledge to fit the current situation and finally, *retain* the knowledge learnt from this problem-solving process. The cases of the RT-CBR module are structured as follows:

$$C = \langle I, F, N_t, N_s, T \rangle \quad (1)$$

where I and F represent the coordinates of a path from the initial position I to the final position F that the robot travelled (one or several N_t times) straight ahead in the past, N_s stands for the number of times that the robot successfully completed the path within the case-based estimated time and T shows the series

of time values that the robot spent to cover that route. Note that only straight routes are stored as cases, since we assume that it is the quickest way between two points. This design decision should minimise the time for travelling an unvisited route that the RT-CBR module would try to compose by reusing several known routes (old cases).

The RT-CBR reasoning cycle starts when the TCA module must decide if an agent can fulfil a shipment service within the time assigned to do it. Next, the operation of each reasoning phase of the module is shown.

Retrieval and Reuse

Due to the temporal constraints that the CBR process has to keep, we have followed an anytime approach [2] in the design of the algorithm that implements the retrieval and reuse phases of the RT-CBR module. In our design, both phases are coupled in the algorithm, reusing the time estimations about several paths to retrieve the most suitable case(s) to travel current routes (the composition of cases that minimises the travelling time). At the end of each iteration, the algorithm provides the manager with a probability of being able to perform the shipment service on time. If more time is still available, the algorithm computes better estimations on subsequent iterations.

First, the RT-CBR module searches its case-base to retrieve a case that represents a similar path that the Robot Agent travelled in the past. Then, for each retrieved case, the algorithm uses a *confidence function* to compute the probability of travelling from an initial to a final point in an area without diverting the agent's direction. Shortest paths are assumed to have less probability that an unpredictable circumstance could deviate the agent from its route and hence, they are preferred from longer ones. In the best case, there will be a case in the case-base that exactly or very approximately covers the same path that the agent must travel. Then, the necessary time to perform the shipment can be estimated by using the time spent in that previous case. Otherwise, the route could be covered by aggregating a set of cases and estimating the global time by adding the time estimation for each sub-case. If the route can be somehow composed with the cases of the case-base, the following confidence function will be used:

$$f_{trust}(i, j) = 1 - \frac{dist_{ij}}{maxDist} * \frac{N_s}{N_t} \text{ where } dist_{ij} \leq maxDist \quad (2)$$

where $dist_{ij}$ is the distance travelled N_t times between the points $\langle i, j \rangle$, N_s represents the number of times that the robot has travelled the path within the case-based estimated time and $maxDist$ specifies the maximum distance above which the agent is unlikely to reach its objective without finding obstacles.

In the worst case, the agent will have never travelled a similar path and hence, it cannot be composed with the cases stored in the case-base. Then, a confidence function that takes into account the distance that separates both points will be used:

$$f_{trust}(i, j) = \begin{cases} 1 - \frac{dist_{ij}}{const_1} & \text{if } 0 \leq dist \leq dist_1 \\ 1 - const_2 * dist_{ij} & \text{if } dist_1 < dist \leq dist_2 \\ \frac{dist_{ij}}{dist_{ij}^2} & \text{if } dist_2 < dist \end{cases} \quad (3)$$

where *const1* and *const2* are normalisation parameters defined by the user, $dist_{ij}$ is the Euclidean distance between the initial and final points of the path $\langle i, j \rangle$ and $dist_1$ and $dist_2$ are distance bounds that represent the thresholds that delimit near, medium and far distances from the initial point. This function computes a smoothed probability that the robot can travel its path straight ahead. As the distance between the initial and the final point increases, the confidence on travelling without obstacles decreases.

Once the probability to reach the robot's objective is computed for each case, the complete route from the initial to the final position with the maximum probability of success must be selected. This route is composed by using a selection function $F(n)$ (4), which follows an A^* heuristic search approach [3]. The function consists of two sub-functions: $g(n)$ (5) that computes the case-based confidence of travelling from the initial point to a certain point n and $h(n)$ (6) that computes an estimated confidence of travelling from the point n to the final point (always better than the real confidence). Finally, the function $T(n)$ (7) checks if the Robot Agent has enough time to complete the shipment service by travelling across this specific route. Else, the algorithm prunes the route. The function consists of two sub-functions: $time(n)$ (8) that computes the case-based time of travelling from the initial point to a certain point n and $E(n)$ (9) that computes an estimated time of travelling from the point n to the final point. In (8) $dist_{mn}$ represents the distance between the last point m visited by the algorithm to the current point n , V_{robot} is the speed of the robot, $f_{trust}(m, n)$ corresponds to (2) or (3) (depending on the possibility of composing the route by using the cases of the case-base) and the constant $const_{trust} \in [0, 10]$ shows the caution degree of the robot agent. Bigger values of this constant stand for more cautious agents.

Finally, if the RT-CBR algorithm is able to compose the entire route with the information stored in the case-base, it returns the case-based probability to perform the shipment service on time. Otherwise, it returns the product of the probability accumulated to that moment and a pessimistic probability to travel from the last point that could be reached by using the cases of the case-base to the final point of the route. Finally, in case that all possible solution computed by the algorithm exceed the time assigned to fulfil the service, it returns a null probability to perform the service successfully.

$$F(n) = g(n) * h(n) \quad (4)$$

$$g(n) = g(m) * f_{trust}(m, n) \quad (5)$$

$$h(n) = 1 - \frac{dist_{nf}}{maxDist} \text{ where } dist \leq maxDist \quad (6)$$

$$T(n) = time(n) + E(n) \quad (7)$$

$$time(n) = time(m) + \frac{dist_{mn}}{V_{robot}} + \frac{const_{trust}}{f_{trust}(m, n)} \quad (8)$$

$$E(n) = \frac{dist_{nf}}{V_{robot}} \quad (9)$$

The probability returned by the RT-CBR algorithm will be used to determine whether the agent can commit itself to performing the service, or whether it must

reject the service. Each agent has a *confidence value*. If the returned probability is greater or equal than the confidence value, then the service will be accepted for execution. This confidence value is different according to the agent behaviour. A cautious agent will have a high confidence value and thus, it will only accept those services with a high probability to fulfil the goal. Instead, a fearless agent will have a low confidence value.

Revision

Once the Robot Agent has finished the shipment service, it reports to the TCA module the coordinates of each path that it finally travelled straight ahead and the time that it spent in doing so. In that way, the manager can check the performance of the RT-CBR module by comparing the time estimated by the module and the real time that finally took the service. Note that if the agent has not changed its navigation algorithm, it will likely try to perform the shipment by following the same route that ended in a success in the past. However, due to that some new obstacles could be found during the route, the design decision of reporting the specific paths that the Robot Agent has travelled has been taken.

Retention

The last step of the reasoning cycle considers the addition of new knowledge in the case-base of the RT-CBR module. As pointed out before, the size of the case-base must be controlled and therefore, only useful cases must be added (and correspondingly, out-of-date cases must be eliminated). Therefore, the decision about the addition of a new case in our model is crucial. At the moment, we have taken a simple but effective procedure by defining a threshold α below which two points must be considered as nearby in our shipment domain. Let us consider a new case c with coordinates (x_i^c, y_i^c) (initial point) and (x_f^c, y_f^c) (final point) to be added in the case-base. Following an *Euclidean* approach, the distance (dissimilarity) between the case c and each case z of the case-base can be computed with:

$$dist(c, z) = \max(\sqrt{(x_i^c - x_i^z)^2 + (y_i^c - y_i^z)^2}, \sqrt{(x_f^c - x_f^z)^2 + (y_f^c - y_f^z)^2}) \quad (10)$$

Therefore, the new case will be included in the case-base iff:

$$\forall z \in caseBase / dist(c, z) < \alpha \quad (11)$$

In that case, the new case $\langle (x_i^c, y_i^c), (x_f^c, y_f^c), 1, 1, time \rangle$ will be added in the case-base (1 values stand for this first time that the path has been successfully travelled). Note that the addition of new cases is always conditioned at the existence of ‘free space’ to add cases in the case-base. Otherwise, a maintenance cycle will be triggered, deleting, for instance, those old cases that have low usages. Else, if a similar case in the case base has been identified, the number of times that the agent has travelled the path that represents the case (N) will be increased by 1 and the time spent in travelling the current path will be added to the time series of that case.

To develop and execute the postman multi-agent system, we have use the JART platform [4] (which is specially designed for systems of this type) and RT-Java [5] as the programming language. Once the example was implemented, several simulation experiments were conducted to evaluate different parameters in order to verify the use of the proposed module. A simulation prototype was im-

plemented using a Pioneer 2 mobile robot simulation software (specifically, the Webots simulator [6]). Next section presents the results of the experiments.

4. Results and conclusions

The simulation experiments were conducted to evaluate different aspects and to try to show the benefits of the CBR integration in an agent with real-time constraints. The different experiments were tested on a system without the TCA module and on one that included the module.

The Floor Agent is informed about the arrival of new mail through an Interface Agent running on a PDA or a mobile phone. The Floor Agent decides the most appropriate Robot Agent to perform the request. Later, the selected Robot Agent is informed of new mail orders. At this point, the Robot Agent must decide whether it accepts the mail order. If it accepts, the Robot Agent is committed to doing the request in the agreed time. If it does not accept, the Floor agent must find another Robot Agent that can do the request.

It is important to note that, in the case of the agent without the TCA module, the received mail orders are stored in a request queue. This queue only stores up to five pending requests. When the Robot Agent receives a request, if it has space in the queue, the Robot Agent accepts the commitment associated to the request. Otherwise the request is rejected. In each case, the mail or package distribution must be ended before a maximum time, and the robot control behaviour must guarantee the robot integrity, which implies hard real-time constraints. In the tests carried out, all requests have the same priority. Therefore, to use the number of requests managed by the Robot Agent is an adequate metric to verify the improvement offered by the use of the TCA module. Thus, the greater the number of requests satisfied on time by the Robot Agent, the better performance the system has. If the requests have different priorities, this metric is not correct. In this case, to fulfil tasks with high priority is more important than to fulfil a greatest number of low priority tasks.

The first set of experiments investigates the request acceptance of the system according to the package or mail arrival frequency to an agent without the temporal constraints analysis (TCA) module and to other agent with the TCA module (for this agent, three test modifying the confidence values from 70%, 80% and 90% are executed). The simulation prototype was tested by increasing this frequency incrementally and by testing the number of unaccepted requests. The tests consisted of groups of 10 simulations of five minutes duration. The Floor Agent received between 5 and 30 requests during these five minutes. Each experiment was repeated one hundred times and the results show the average value obtained. The results are shown in Figure 2. This figure shows that, at a low request frequency, the Robot Agent accepts all requests and is committed to fulfilling them, independently of whether the Robot Agent incorporates the TCA module. Nevertheless, when the request frequency is increased, the agent with the module works better (independently of the confidence value), even with a high frequency rate. These results must be contrasted with the percentage of successfully completed requests. With respect to the agent without the module, it can be

observed that, in the case of low average rates, the behaviour is slightly better at the beginning. The reason behind is that the agent accepts all requests as long as its request queue is not full, without taking into account whether it will be able to successfully process these requests. The Robot Agent that has the TCA module rejects requests sooner than the other one because it only accepts requests if it is able to successfully complete the work associated with that commitment. The

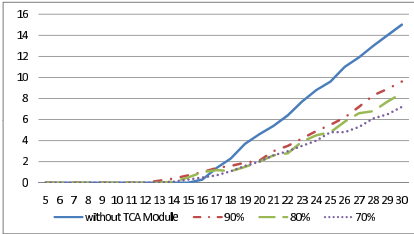


Figure 2. Analysis of the number of unaccepted requests

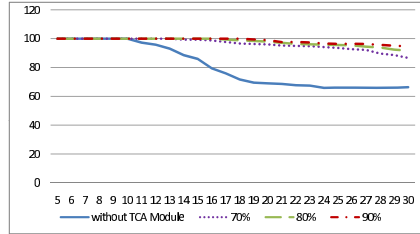


Figure 3. Percentage of successfully ended requests

second set of experiments investigates the success rate of accepted commitments according to package or mail arrival frequency (Figure 3). This figure shows that using the TCA module with a confidence value of 90 % is very efficient at maintaining the success rate close to 100%. In contrast, if the Robot Agent does not use the module, the success rate decreases as the number of requests increases. Even when the saturation of requests in the system is very high, the agent with the TCA module still had a success rate of approximately 90% independently of the confidence value. When the confidence value decreases the success rate is worse. The reason for this loss is the time consumed by the TCA module to analyse the requests and to control the execution of the accepted request. At the beginning, this execution time is bounded, but when the request arrival frequency is very high, this time is overrun, which affects the execution of the rest of the agent's tasks. This will be improved in next implementations of the TCA module.

In conclusion, the proposed module allows a suitable management of the bounded requests that real-time agents acquire from other agents. With the TCA module presented in this paper, it is possible to guarantee that the Robot Agent works correctly according to their commitments in a soft real-time system, as the postman problem described is.

References

- [1] Aamodt, A. and Plaza, E., Case-based reasoning; Foundational issues, methodological variations, and system approaches. *AI Comm.*, vol. (7)1: 39-59, 1994.
- [2] Dean, T., Boddy, M., An analysis of time-dependent planning. *Proceedings of the seventh National Conference on AI*, pp. 49-54, Minnesota, 1988.
- [3] Hart, P.E., Nilsson, N. J., Raphael, B., A formal basic for the heuristic determination of minimum cost paths. *IEEE Transactions on SSC* 4:100-107. 1968
- [4] M. Navarro, V. Julián, Jose Soler, V Botti: jART: A Real-Time Multi-Agent Platform with RT-Java. In *Proc. 3rd IWPAAMS'04*, pp. 73-82, 2004.
- [5] RTJ: The Real-Time for Java Expert Group. <http://www.rtlj.org>
- [6] Webots: <http://www.cyberbotics.com/>

Representing Qualitative Trajectories by Autonomous Mobile Robots

J. C. Peris¹, J. Plana and M. T. Escrig²

Universitat Jaume I

¹*Languages and Computer Science Systems Department*

²*Engineering and Computer Science Department*

Campus Riu Sec, Castellón, E-12071 (Spain)

jperis@lsi.uji.es, al004104@alumail.uji.es, escrigm@icc.uji.es

Abstract. Description of trajectories has been usually dealt in quantitative terms by means of mathematical formulas. Human beings do not use formulas to describe the trajectories of objects in movement, for instance, for shooting or catching a ball. Common sense reasoning about the physical world has been commonly solved by qualitative models. In this paper, we are going to qualitatively describe trajectories of objects in movement, at constant speed and in a 2D space, by an AiboTM robot.

Keywords: Spatial Reasoning, Qualitative Reasoning, Qualitative Trajectories, Autonomous Mobile Robots

Introduction

A new technique for a cognitive description of the movement of objects for service robotics is necessary. Up to now, most of the approaches developed to describe trajectories are quantitative: the quantitative model to plan the motion of mobile robots (determination of the no collision trajectories) developed in [1] has been used to maintain a certain formation class of mobile robots using a geometric approach [2], and to define a strategy leader-follower approach applied to the planetary exploration [3]. Another quantitative approach to define a near-time-optimal trajectory for wheeled mobile robots has been developed in [4]. A quantitative model for planning the motion of parallel manipulators has been developed in [5].

However, few works exist in the literature which use qualitative representations: in [6] a qualitative representation of orientation combined with translation for a parallel manipulator for diagnosis is presented. To date, collision detection methods have been totally quantitative, obtaining much more details than strictly necessary at a high computational cost [7].

On the other hand, human beings do not use formulas to describe trajectories of objects in movement, for instance for shooting or catching a ball. We use the common sense reasoning. Qualitative Spatial Reasoning (QSR) is, perhaps, the most common way of modelling commonsense reasoning in the spatial domain. A qualitative representation can be defined as that representation which makes only as many distinctions as necessary to identify objects, events, situations, etc. in a given context. Qualitative models are suitable for further reasoning with partial information in a

robust manner, in cases where complete information is either unavailable or too complex to be efficiently managed.

In this paper, we are going to qualitatively describe trajectories of objects in movement by a robot. The remainder of this paper is organized as follows. Section 1 presents the kinematics aspects which are used to qualitatively describe trajectories. In section 2 we will show the qualitative representation constructed by an AiboTM robot. In section 3, the conclusions and prospects are explained.

1. Qualitative Representation of Trajectories by Autonomous Mobile Robots

For representing the kinematical properties of objects in movement it is necessary to introduce the concept of trajectory. A trajectory can be considered as the representation of the different dynamical states of a moving object (figure 1).

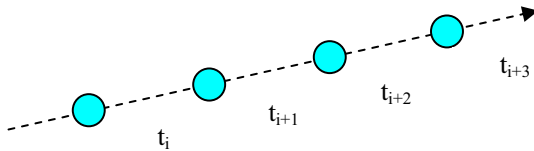
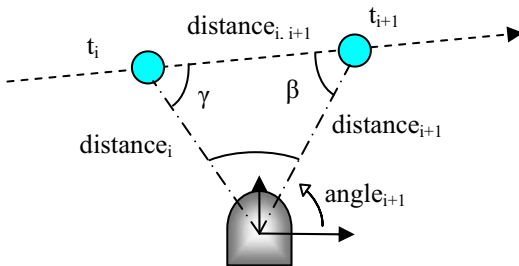


Figure 1. Representation of a trajectory as the successive positions of an object at different times.

It is important to remark that this approach has been developed considering a uniform movement. The presence of any kind of acceleration will involve other geometric considerations. Moreover, the model considers only trajectories in a 2D space.

Each object of the environment detected by the robot is positioned with respect to the robot orientation. The front of the robot is considered the 90° angle and positive angles are measured to the left of the robot. The information stored can be viewed as a pair of elements representing a polar coordinate $\langle d_i, a_i \rangle$, where d_i is the distance from the centre of the robot to the object and a_i is the angle from the front of the robot.

With the detection of the position of an object at two different times, t_i and t_{i+1} , we can obtain the distance between them, $d(i, i+1)$, and the angles α, β, γ formed by these positions and the robot, by means of the triangulation technique, using the cosine and sine formulas (figure 2).



$$d(i, j)^2 = d_i^2 + d_j^2 - 2 \cdot d_i \cdot d_j \cdot \cos(\alpha)$$

$$d(i, j) / \sin(\alpha) = d_i / \sin(\gamma) = d_j / \sin(\beta)$$

Figure 2. Geometrical information obtained from 2 different positions of a moving object at times t_i and t_{i+1} .

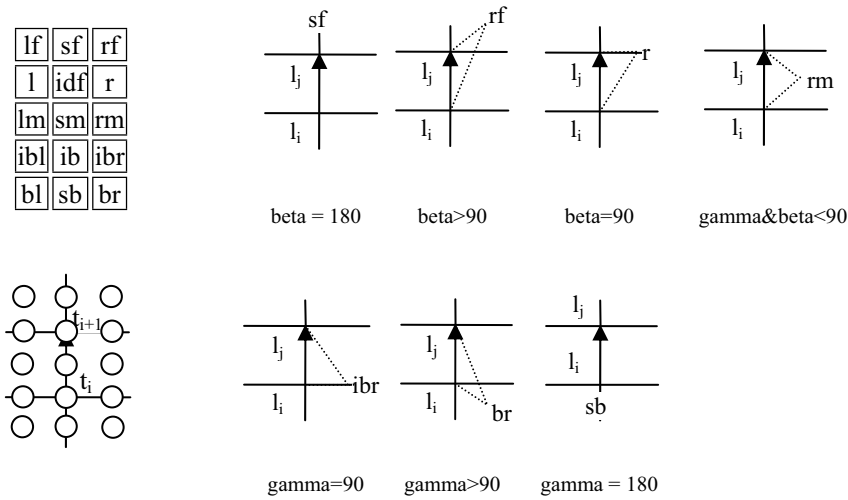


Figure 3. The qualitative orientation model [8].

Figure 4. Seven possible orientations of the robot wrt two different positions of a moving object at times t_i and t_{i+1} . We only consider one side (the side of the robot) of the reference system.

Moreover, we use the [8]’s qualitative orientation model (figure 3) for describing the qualitative orientation of the robot with respect to the reference system formed by two positions of the moving object (figure 4).

As a matter of example, the robot orientation with respect to (wrt) the reference system formed by the moving object in two instants of time t_i and t_{i+1} , in figure 2, is *rm* (right middle).

We also want to represent the movement of an object with respect to the robot orientation, therefore we overlap the [9]’s orientation reference system (figure 5) to the center of the robot. This reference system divides the space into qualitative regions. The number and size of these regions depends on the granularity we want to use. It might be application dependent.

This method is still valid when the robot is moving. In that case we are calculating relative movements between the robot and the objects.

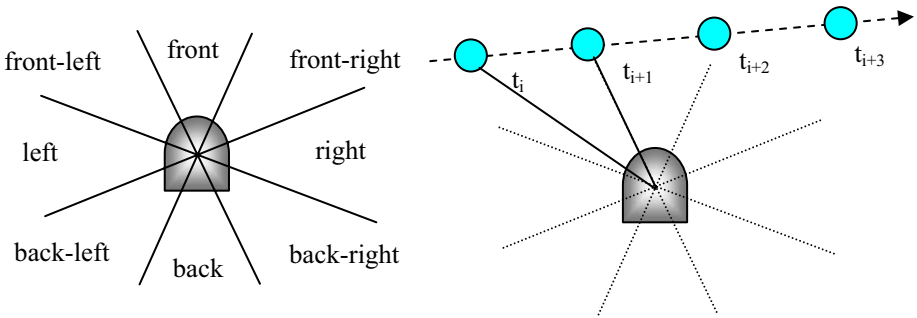


Figure 5. Space division using [9]’s orientation model.

Figure 6. We can see how the moving object goes from front-left to front and from front to front-right orientations.

| rob. wrt t_i, t_{i+1} | object wrt robot = FRONT | rob. wrt t_i, t_{i+1} | object wrt robot = BACK |
|----------------------------|---|----------------------------|--|
| lf | Object goes from front to left side | lf | Object goes from back to right side |
| l | Object goes from front to left side | l | Object goes from back to right side |
| lm | Object goes from front to left side | lm | Object goes from back to right side |
| ibl | Object goes from front to left side | ibl | Object goes from back to right side |
| bl | Object goes from front to left side | bl | Object goes from back to back-right |
| rf | Object goes from front to right side | rf | Object goes from back to left side |
| r | Object goes from front to right side | r | Object goes from back to left side |
| rm | Object goes from front to right side | rm | Object goes from back to left side |
| ibr | Object goes from front to right side | ibr | Object goes from back to left side |
| br | Object goes from front to right side | br | Object goes from back to back-left |
| sb | Object is in front side & is going away | sb | Object is in back side & is going away |
| sf | Object is going to collide | sf | Object is going to collide |
| else | No movement | else | No movement |

Figure 7. Two of the eight tables representing the knowledge about qualitative trajectories

Qualitative trajectories are calculated by means of eight tables, one for each one of the orientation where an object can be seen in t_i in the [9]'s RS. Two tables, when the object is located to the FRONT / BACK of the robot, are shown in figure 7. The rest of the tables are calculated in a similar way and are not included due to space restrictions. In these tables, a relationship has been obtained between:

- Qualitative orientation of the robot with respect to the moving object (figures 2, 3 and 4): robot wrt t_i, t_{i+1}
- Qualitative orientation of the current position of the moving object with respect to the robot orientation (figure 6): object wrt the robot.

2. Qualitative description of trajectories by an Aibo robot

We have used this methodology for the qualitative representation of the trajectory described by a pink ball which has been thrown to an AiboTM robot.

AiboTM is a four-legged dog-like entertainment robot developed by Sony. It has a number of sensors and actuators, including a colour camera, which can be manipulated by software. In addition, we use the commercial mobile robot simulation WebotsTM, developed by Cyberbotics¹ Ltd. The WebotsTM software is shipped with a simulation of the AiboTM robot allowing the use of a cross-compilation process, wherein the controller code is cross-compiled to produce a binary executable which then runs on the live robot directly. We have added to the WebotsTM cross-compiler the necessary functions for using the robot camera [10], since this capability was not implemented. Therefore, we can develop and test our controllers on the simulator before executing them on the real robot (figure 8).

The complete algorithm for the extraction of the qualitative trajectories is composed by the next steps:

1. Image acquisition and colour segmentation.
2. Object localization and tracking.
3. Calculating the distance between the robot and the ball.
4. Representing the trajectory of the ball.

¹ www.cyberbotics.com

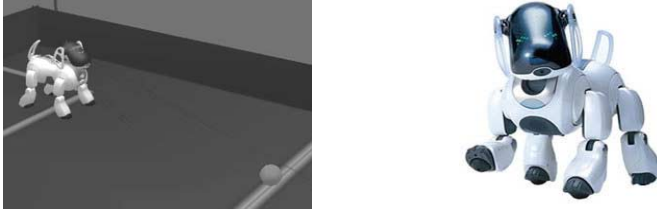


Figure 8. a) left: a Webots simulated environment with an Aibo robot; b) right: a real Aibo robot.

2.1. Image acquisition and colour segmentation

The two initial steps we use in this work for extracting trajectories are image acquisition, by means of the robot camera, and colour segmentation. The AiboTM camera has a resolution of 208x160 pixels and we use the YUV colour space. We detect objects in images using colour blob detection [11]. Colour blob detection is a method to find objects by their colour. In our tests pink balls are searched. The output of the process is binary data for each pixel, indicating whether the pixel belongs to the ball or not.

We use a threshold to filter the colours of objects we are looking for. Threshold values define the range of colour values that classify a pixel to belong to a blob. A simple threshold calculus is sufficient when images come from an environment with homogeneous light. This was not our case, then a simple threshold calculus yields mixed results because colours are too diverse and do not fall within a static threshold range. To solve this problem we have developed a transformed threshold calculus which consists in two steps. In the first step, a simple threshold calculus is performed with a relatively narrow threshold range. Only the centre part of the ball, which does not suffer from over illumination or shade, will be classified as belonging to a blob in this phase. The second step consists in a growing process on every pixel that was classified as belonging to a blob. We relax the threshold. We look at the surrounding pixels of each pixel chosen in the first step, if they have similar threshold, then this pixel is added to the newly formed blob. When a new pixel is added to the blob, a new average of the Y, U and V colour channels is calculated, which corresponds to the new threshold. Neighbouring pixels of this pixel are compared to this average instead of the colour of the pixel itself.

The formulas below illustrate the defined growing conditions. In the formulas, $y_{average}$, $u_{average}$ and $v_{average}$ stand for the average y, u and v values. The colour that is checked to see if it is similar is defined as y_{pixel} , u_{pixel} and v_{pixel} . There are three threshold values, one for each colour channel. They are defined as t_y , t_u , t_v .

$$y_{average} = 1/n \sum_n y_i \quad u_{average} = 1/n \sum_n u_i \quad v_{average} = 1/n \sum_n v_i$$

$$(|y_{pixel} - y_{average}| < t_y) \wedge (|u_{pixel} - u_{average}| < t_u) \wedge (|v_{pixel} - v_{average}| < t_v) \quad (1)$$

A blob is represented as a square with the dimensions x_{min} , y_{min} , x_{max} , y_{max} , which are the minimum and maximum x and y positions of the pixels inside the blob. Besides this square, the gravitation point (g_x , g_y) and the size in pixels of the blob are calculated.

2.2. Object localization and tracking

Once the robot starts execution, it moves the head from side to side taking images and looking for the ball. This movement is repeated until a pink blob (the colour of the ball)

is extracted from the colour segmentation process. When the ball is found the robot has to track it. This is done by moving the head to situate the ball in the centre of the camera image. We use the next formulas and functions:

$$\begin{aligned} pos_x &= pos_x + ((ImageWidth / 2) - g_x) / n \\ pos_y &= pos_y + ((ImageHeight / 2) - g_y) / n \end{aligned} \quad (2)$$

servo_set_position(head_pan, pos_x)
servo_set_position(head_tilt, pos_y)

ImageWidth and ImageHeight are the size of the camera, 208 and 160 respectively. The variables pos_x and pos_y are the position in radians of the head servos *head_pan* and *head_tilt*, respectively. The values of these variables are actualized based on the previous ones, but the first time the ball is found these variables are initialized to the position of the head servos at that moment. The variables g_x and g_y are the components of the gravitation point. The *servo_set_position* function moves the head servos to the position indicated by the values of the pos_x and pos_y variables. Finally, n determines the conversion between pixels and radians.

The value of the variable n has been calculated both in WebotsTM and the real robot executing a test process where the robot moves the head servos a number of radians and calculates the displacement in pixels of the ball in the image. This value depends on the distance between the camera and the ball. We have worked with distances about 1.5 and 2 meters. The results for the real robot are showed in the table 1. We can see that the value obtained for the *head_pan* servo is aprox. 4 and the value for the *head_tilt* servo is aprox. 3. But using these values produce oscillations in the head movement due to the acceleration of the servos. The final value has been chosen when no oscillations are produced. See table 2 for the values tested on the *head_pan* servo of the real robot.

| Degrees | Pixels | Degrees | Pixels |
|-----------------|--------|------------------|--------|
| 3 | 10 | 3 | 10 |
| 4 | 14 | 4 | 14 |
| 5 | 20 | 5 | 18 |
| 6 | 26 | 6 | 18 |
| 7 | 28 | 7 | 21 |
| 8 | 32 | 8 | 24 |
| 9 | 36 | 9 | 26 |
| 10 | 40 | 10 | 30 |
| <i>head_pan</i> | | <i>head_tilt</i> | |

Table 1. Values of n for the real robot.

| n | Servo behavior | Convergence |
|-----|-----------------|-------------|
| 3 | oscillations | no |
| 4 | oscillations | no |
| 5 | oscillations | yes |
| 7,5 | no oscillations | yes |
| 10 | no oscillations | slow |

Table 2. Behavior of the *head_pan* servo for different values of n .

Convergence indicates if the ball is centred or not in the image. The final value of n for all the sensors is therefore 7.5. The same value has been obtained for the WebotsTM simulation.

2.3. Calculating the distance between the robot and the ball

Every time the ball is centred in the image the distance between the robot and the ball is calculated. The method we use calculates the distance by means of relating some characteristics of the blob detected by the camera with the real diameter of the ball [12]. The equations used are:

$$\begin{aligned} distance &= FocalLength \cdot (BallDiameter / D) \\ D &= \max(Rv, Rh, (\sqrt{S} \cdot 2 / \pi)) \end{aligned} \quad (3)$$

where FocalLength and BallDiameter are intrinsic parameters. FocalLength is the focal length of the camera (192mm), BallDiameter is the real diameter of the ball (70mm), Rv is the vertical length of the blob, Rh is the horizontal length of the blob, and S is the size of the blob in pixels. The Rv, Rh and S values are obtained from the information provided by the colour segmentation process:

$$Rh = xmax - xmin \quad Rv = ymax - ymin \quad S = size \quad (4)$$

The error obtained with this method with good light conditions is small because it only depends, if the intrinsic parameters are correct, on the segmentation method. The average error obtained in our tests can vary between 5 and 10 centimetres. Theoretical studies of the error, using the depth and assuming errors in the segmentation, will be made and compared.

2.4. Representing the trajectory of the ball

As we have seen in previous subsections, we have been dealing with quantitative information extracted from the robot camera. This quantitative information is necessary in a first level for extracting relevant information from the environment. There are two relevant values which will be useful for identifying qualitative trajectories: the *distance* from the robot to the ball and the position of the *head_pan* when the ball is centred in the camera image. By taking two pair of values (distance, head_pan) at two different times, we can identify the qualitative trajectory of the ball applying the methods explained in section 3. This is done continuously by the robot, identifying possible changes in the trajectory of the ball. Furthermore the calculus is obtained each three images because the difference of distances and angles can not be small.

Tests obtained with WebotsTM simulation have a 100% of success in trajectory identification. We use in our simulations perfect light conditions and no errors in the sensors measurements. Tests realized with the real robot have a 90% of success in trajectory identification. The errors were produced mainly by poor blob detection due to bad light conditions. Another source of errors is the accuracy of the head servos of the robot. When the displacement of the ball in successive images is small, the movement of the correspondent head servo is small too, producing bad positioning.

3. Conclusions and Future Work

Description of trajectories has been traditionally dealt in quantitative terms by means of mathematical formulas. Human beings do not use formulas to describe the trajectories of objects in movement, for instance, for shooting or catching a ball. Common sense reasoning about the physical world has been commonly solved by qualitative models. However, no qualitative model about trajectories of objects in movement has been developed up to our knowledge.

We have seen in this paper a process for representing qualitative trajectories by an autonomous mobile robot. We have been dealing with quantitative information extracted in this case from a camera. This quantitative information is necessary in a first level for extracting relevant information from the environment. There are two relevant values which will be useful for identifying qualitative trajectories: the distance from the robot to the moving object and the angle of the object with respect a reference

system centred in the robot. By taking these two pair of values (distance, angle) at two different times, we can identify the qualitative trajectory of an object. This is done continuously by the robot, identifying possible changes in the trajectories. Tests obtained with WebotsTM simulation make successfully qualitative trajectory identification. In this sense, this description is able to grasp an important kinematical aspect such as the collision condition.

The work presented in that article is focused on the qualitative relationship between a moving object and the robot, which we think could be integrated inside the construction and representation of cognitive maps. Future work about this qualitative representation can be applied to robotics in the following main aspects:

- Collision detection.
- Qualitative description of dynamic worlds for map building.

Moreover, more work is needed with real robots to extract a complete set of results. We have to study the impact of wrong measures in the description of qualitative trajectories.

Acknowledgements

This work has been partially supported by CICYT under grant number TIN 2006-14939.

References

- [1] J.C. Latombe. Robot Motion Planning, Kluwer, 1991.
- [2] C. Belta, V. Kumar. Motion generation for formation of robots ,In Proceedings of the 2001 IEEE International Conference on Robotics and Automation, Seoul, Korea, 2001.
- [3] T.D. Barfoot, E.J.P. Eaton, G.M.T. D  leuterio. Development of a network of mobile robots for space exploration. The 6th International Symposium on Artificial Intelligence, Robotics and Atomation in Space, 2001.
- [4] J.S Choi and B.K. Kim. Quantitative derivation of a near-time-optimal trajectory for wheeled mobile robots. IEEE Transactions on Robotics and Automation, vol. 17, no. 1, february 2001.
- [5] Merlet J.P.. Direct kinematics of parallel manipulators. Robotics and Automation, IEEE Transactions on Volume 9, Issue 6, Dec. 1993.
- [6] Liu, H., Goghil G.M. Qualitative Representation of Kinematic Robots. Proceedings of the 19th International Workshop on Qualitative Reasoning, Graz, Austria. 2005.
- [7] Ericson, C., Real-Time Collision Detection. The Morgan Kaufmann Series in interactive 3-D Technology. Ed. Elsevier. ISBN: 1-55860-732-3, 2005.
- [8] Freksa, C., Zimmermann, K., "On the Utilization of Spatial Structures for Cognitively Plausible and Efficient Reasoning", Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, pag. 18-21, 1992.
- [9] Hern  ndez, D. Qualitative Representation of Spatial Knowledge. In volume 804 in Lecture Notes in AI. Ed. Springer-Verlag, 1994.
- [10] J.C. Peris, J. Grande and M.T. Escrig. Adding camera functions to the Webots OPEN-R wrapper object for Aibo robots. Technical report ICC 2008-05-02, Engineering and Computer Science Department, Jaume I University.
- [11] Michiel Punter and Tjitze Rienstra *The Slam Challenge Project*, tech. rep., Computer Science at the Noordelijke Hogeschool, Leeuwarden, 2005.
- [12] H. Fukumoto, T. Nagatomi, S. Oono, K. Hino, H. Kawai, W. Uemura, N. Mitsunaga, A. Ueno, S. Nakajima, S. Tatsumi, T. Toriu, and M. Asada. *BabyTigers DASH: Osaka 4-Legged Robot Team*, tech. rep., Osaka City University and Advanced Telecommunications Research Institute International and Osaka University, 2006.

Object-based Place Recognition for Mobile Robots Using Panoramas

Arturo RIBES ^{a,1}, Arnau RAMISA ^a and Ramon LOPEZ DE MANTARAS ^a and
Ricardo TOLEDO ^b

^a *Artificial Intelligence Research Institute (IIIA-CSIC), Campus UAB, 08193 Bellaterra, Spain*

^b *Computer Vision Center (CVC), Campus UAB, 08193 Bellaterra, Spain*

Abstract. Object recognition has been widely researched for several decades and in the recent years new methods capable of general object classification have appeared. However very few work has been done to adapt these methods to the challenges raised by mobile robotics. In this article we discuss the data sources (appearance information, temporal context, etc.) that such methods could use and we review several state of the art object recognition methods that build in one or more of these sources. Finally we run an object based robot localization experiment using an state of the art object recognition method and we show that good results are obtained even with a naïve place descriptor.

Keywords. Object recognition, Localization, Mobile robots

Introduction

Object recognition² is one of the most promising topics of Computer Vision. Having the ability to learn and detect hundreds of arbitrary objects in images from uncontrolled environments would be a major breakthrough for many artificial intelligence applications. One of the areas of research that would benefit the most of that innovation is intelligent robotics. The capacity to perceive and understand the environment is an important limitation for designing robots suitable for being deployed in houses to perform in domestic tasks or help the disabled. Also, being able to communicate at human-level semantics with its owners would make these robots much easier to control for a non-technical end user.

In this work we review some of the most relevant state-of-the-art object detection and classification methods. Also, we discuss which sources of information can be exploited in order to have a fast and robust object recognition method, and which of the existing techniques could be applied in such domain. An area of research that recently started to profit from object recognition capabilities is topological localization. In [16] an object-based robot navigation and place categorization method is proposed.

¹Corresponding Author: Arturo Ribes, Artificial Intelligence Research Institute (IIIA-CSIC), Campus UAB, 08193 Bellaterra, Spain; E-mail: aribes@iiia.csic.es.

²Object recognition is used loosely to embrace the concepts of detection, identification and eventually classification.

To illustrate our interest for object recognition methods for topological localization, we perform an object-based localization experiment using a publicly available dataset of stitched panoramic images acquired with a mobile robot. For this experiment, a custom implementation of the Lowe object recognition algorithm [3] has been used to learn and detect the instances of forty different objects in the panoramic images.

The paper is divided as follows. In Section 1 the state of the art in visual object recognition is reviewed, paying special attention to those methods and strategies that adapt well to the domain of mobile robotics. In Section 2 the Lowe object recognition algorithm is explained. In Section 3 the experiments performed are described and the results presented. Finally, in Section 4 the conclusions are drawn.

1. State of the Art

Local features are widely used in computer vision due to its relation with research in primate vision. Those features are regions in an image providing relevant information used as building blocks of the models that characterize an object. This bottom-up approach to vision enables the models to be stable against background clutter and occlusions, while properties of local descriptors account for illumination and viewpoint invariance. Feature sampling strategies commonly used are based in interest region detectors, which finds regions in a well-founded mathematical way that is stable to local and global perturbations. Using this techniques texture-rich regions can be detected at repeatable positions and scales.

1.1. Object Classification

Recently significant work has been done in visual object classification, with many methods making analogies to document retrieval literature. Visual vocabularies have been extensively used to relate local feature regions to visual words by clustering its descriptors with algorithms like k-means [14], hierarchical k-means [8] or agglomerative clustering [2], among others. This reduces its dimensionality to a fixed number, so visual-word statistics can be used to classify the input image.

In [14] the authors use k-means to cluster local feature descriptors into a vocabulary of visual words and the TF-IDF (Term Frequency - Inverse Document Frequency) scheme to prioritize distinctive feature descriptors. The intuition is that TF term weights words occurring often in a particular document, and thus describe it well, whilst IDF term downweights words that appear often in the database. Local features are detected in an image with an affine-invariant adaptation of Harris corner detector [5] and MSER detector [4] and a histogram is built from visual-word counts. Experiments are done in scene matching - where query descriptor vectors are compared to the ones in database - and object retrieval throughout a movie, where a user-specified query region is used to re-rank the results using spatial consistency, accomplished by matching the region visual-words to the retrieved frames.

Similarly [8] uses the same TF-IDF scoring scheme but this time the vocabulary is constructed with a hierarchical k-means applied to MSER features extracted from the set of training images. A tree with a branch factor of k is computed, where at each child node k-means is applied to the contents of the parent node. The step is repeated until a

desired depth is reached. This allows an efficient lookup of visual words at logarithmic cost, enabling the use of larger vocabularies. The results show how well the system scales to large databases of images in terms of search speed and the larger vocabulary describes much better the images so, in contrast with [14], geometric information is not really needed to obtain good performance, although other feature detectors could have been used, like affine-invariant Harris detector.

The work of Opelt et al. [9] proposes combining multiple feature types in a single model making use of boosting to form each classifier. This enables each category to choose the best features, regardless of its type, that describes it. Local features used are both discontinuity and homogeneity regions. The former are appearance features detected and described with various methods that capture local intensity changes and the later are regions extracted with wavelets or segmentation that contain repetitive textures or stable intensity distributions respectively.

Results show that different feature types perform much better in some classes than others, so combining them in a single model greatly improves classification results. This method does not cluster features into codebooks, but forms weak classifiers made of a feature value and a threshold. These weak classifiers are combined in a boosting framework to get final classifiers. Experiments may be done in order to incorporate visual vocabularies as source of weak classifiers, or replacing TF-IDF scoring scheme by any flavour of boosting, like AdaBoost.

1.2. Object Localization

Although good results have been obtained using *bag of words* type methods, when the object does not occupy the majority of the image or detection and pose estimation is necessary, information on the relative position between object features is essential. This is usually a requirement when dealing with real world images, and several methods have been developed in order to take advantage of this positional information.

Leibe et al. present in [2] the Implicit Shape Model, an object detection method that combines detection and segmentation in a probabilistic framework. A codebook of greyscale local patches around Harris corner points is built using agglomerative clustering. Then we compute the object center relative to every feature in each cluster above a certain threshold and store them. This ends up in a mapping between visual words and possible object centers given the images it has seen so far, an idea exploited by many other methods. For detection, query regions extracted from novel images are matched to the codebook, so each cluster casts votes for object centroid positions using the Generalized Hough Transform.

This strategy differs from bag-of-words models in that it implicitly incorporates localization by using a continuous voting space, which is later filtered to obtain stable object centroid predictions. Segmentation is accomplished by overlapping matched regions, and the process can be iterated searching regions nearby the initial ones detected, obtaining a better detection-segmentation. Finally, the image is segmented assigning to each pixel the most probable object class with respect to the matched patches that include it. Tests have been done with the UIUC car database and with the individual frames of a walking cows video database, reporting very good performance. However, results show that the method only is able to deal with very small scale changes (10% to 15%). To avoid this, authors suggest using scale-invariant interest point detectors or rescaled versions of the codebook.

In [10], Opelt et al. present the Boundary-Fragment Model (BFM). This strategy is similar to the one of [2], but instead of local patches, it uses boundary fragments. A codebook of fragments for detecting a particular class is built by first computing Canny edges of the training images and finding edge segments that match well in a validation set (and bad in the negative examples set) using an optimization procedure. Next the candidate edge segments are clustered using agglomerative clustering on medoids to reduce redundancy, storing also the centroid of the object. Groups of two or three segments that estimate well the centroid of the object are used as weak detectors in a boosting framework. A strong detector is trained selecting weak detectors with good centroid prediction capabilities in positive images and that do not fire in negative images.

For detection, weak detectors are matched to image Canny edges, with each one voting for one or various centroid positions in a 2D Hough voting space. Votes are accumulated on a circular window around candidate points, taking those above a threshold as object instances. Finally approximate segmentation can be obtained backprojecting the segments that voted for a centroid back into the image. In order to make the method robust to scale and in-plane rotation different scaled and rotated versions of the codebook are used simultaneously. Authors extended this method for multi-class object detection in [11] using a shared codebook. The method can be trained both jointly or incrementally to add new classes with less cost, sharing weak detectors from other classes if they have good performance with new class. This is a good feature for mobile robots that have to deal with new environments and also manages complexity in an efficient way.

In [17], authors propose a shape-based method for object detection. The method builds on the ISM and uses k -Adjacent Segments with $k=3$ (TAS) as shape-based feature detector. A codebook of TAS is generated by first building a full connected graph of all the TAS in the training set with distance in the edges between every pair of TAS and then applying Normalized Graph Cuts to obtain the cluster centers.

Similarly to [2], probabilistic votes are casted in a 2D Hough Transform, and Parzen Windows are used to find the most plausible object centers. Finally Gradient Vector Flow is used to find an approximate segmentation of the objects. The proposed method is interesting and has a similar performance to BFM in cows and motorbikes dataset used in [10] although using simpler features, specially in the case of few training images and small codebooks (200 features).

In [3] the author proposes a single-view object recognition method along with the well known SIFT features. First SIFT features of the training images are compared with Euclidean distance to the set of descriptors from the test image using a K-D tree and the Best Bin First algorithm to speed up the matching process. Matches in which the distance ratio between first and second nearest neighbors is greater than 0.8 are rejected. This eliminates 90% of the false matches while discarding less than 5% of the correct matches. Matches remaining after this pruning stage are clustered using its geometrical information in a 4D Generalized Hough Transformation with broad bin sizes, which gives the initial set of object hypotheses. To avoid the problem of boundary effects in bin assignment, each keypoint match votes for the 2 closest bins in each dimension. Bins of the Hough Transform containing more than three matches constitute an object location hypothesis. Finally a least-squares method (IRLS) is used to estimate the affine transformation of the detected object.

1.3. Context in Object Recognition

Context has been argued to provide very relevant information in computer vision. In [9], weak classifiers are selected with a boosting method to form the final classifiers; it was found in the experimental results that a high percentage of weak classifiers with higher weights selected local features corresponding to background. This can be interpreted as that context in some classes is more discriminative than foreground parts, although it greatly depends on the training set used.

Also, context can be introduced in an ontology-based manner between objects as done in [12], where object co-occurrence probability is computed by extracting related terms to each class from Google Sets or directly computing it from the ground truth. The system first segments query images into regions, which are in turn classified in a *bag of words* fashion and results obtained are re-ranked based in object co-occurrences.

In [15] context is provided using global features. First, local features are extracted using a collection of steerable filters that are applied to images. To compute the global feature vector for each image, their local features are averaged over a coarse spatial grid in order to obtain a global descriptor of 384 dimensions, which is finally projected to its 80 principal components. Knowing that video frames captured in the same place have a strong correlation, the authors use a Hidden Markov Model to compute the probabilities of being in a specific place given global features observed in the past frames up to the present frame. Their model also can incorporate local features, but it is left to further research, as they want to focus on global features. Place recognition results are also proven to provide a strong prior for object detection. This method shows a form of time-integration of visual cues, which is another desirable feature in a mobile robot vision system, as it operates with video streams, not solely static frames.

1.4. Object Recognition in Mobile Robots

We are aware of few works that consider the case of general object recognition in the domain of mobile robots. In [1] the authors integrate object recognition and SLAM. The proposed object recognition method works with few training images, and consists of two stages (Hypotheses generation and verification) that use active vision to focus on interesting parts of the image. The first stage uses Receptive Field Cooccurrence Histograms (RFCH) and, in the verification stage, an area of interest that contains a maximal number of hypotheses is zoomed in using the optical zoom of the camera. Again RFCH are used to verify each hypothesis and finally SIFT features are extracted and matched to the model image. No further geometrical constraints are applied to discard outliers, and if an object dependent number of SIFT features are matched the object is considered detected. Background subtraction is used during the training stage to precisely segment the objects.

In [7] the authors present a system that learns and recognizes objects in a semi-autonomous fashion. Their system acquires learning samples of new objects by analyzing a video sequence of a teacher showing the object in a variety of poses and scales. Then it automatically segments and builds an internal representation with a shared vocabulary of visual features. The vocabulary is made using k-means over features based on Gabor-jets and hue-saturation values extracted at Harris corner points. Each visual word stores the relative position of the objects where it appears.

Recognition is done by accumulating votes on a Hough Transform in a similar way as in [2]. In the experiments, the system runs at about 1-2Hz, recognizing objects from a 100 object database with 78% of mean accuracy. Additionally, the database included difficult objects, like untextured, flexible and very similar objects (wires, cans, water bottles). Pose and scale invariance is obtained by superimposing multiple learning samples, so it is only invariant in the range that the object was learned.

2. Object Recognition in Panoramas

The method used for object recognition is the one proposed by Lowe [3] with some improvements (Figure 1 shows an outline of the method) in order to obtain better results given that we use panoramic images from our mobile robot (see Figure 2). We use one training image per object in the library, storing its descriptors along with their object ID in a descriptor database. All training images are taken from a 1Mpx digital camera.

Testing images are panoramas built from 36 images acquired with the pan-tilt camera mounted on top of our mobile robot, storing the set of SIFT descriptors extracted from image regions used to construct the panorama. The position of each descriptor is relative to the total panorama size, not to the individual images. The panoramic image is used only to show recognition results, as the internal representation used is the descriptor set.

Our implementation of the Lowe SIFT object recognition method differs from the original in the keypoint matching method, where instead of one single K-D tree and the BBF algorithm we use the improved method proposed in [6]. This method gives an improvement in search time of two orders of magnitude with respect to linear search and retrieves 90% of true nearest neighbors.

Hough clustering stage is very tolerant to outliers, but only provides a similarity transform as a starting point for the more accurate affine transform estimation. Given that IRLS is very sensitive to outliers, we evaluate the impact of adding an intermediate RANSAC (RANDOM Sample Consensus) stage that provides a good initialization of affine parameters and is also tolerant to outliers.

Furthermore, given our mobile robot setup, we can discard hypotheses with some heuristics applied to the affine solution bounding box related to the panorama size.

- A box that spans more than 90° of field-of-view (25% the width of panorama) is rejected: this would correspond to an object being in touch with the robot, thus not recognizable by its huge scale and viewing conditions.
- A box that is not at least 10% inside of view is rejected because there cannot be sufficient distinguishable features to estimate the affine transform between the object and panorama.
- A box in which the ratio of minor to major axis is less than 0.05 is rejected, corresponding to objects in viewing conditions that are proven not to be recognizable with this method.

Although this heuristics come from human knowledge of the method limitations or the robotic setup used, they can be supported with detector experiments or inferred from localization results and incorporated in a probabilistic model, which is left for further research. We use this method to build a signature for a panorama consisting in the presence or absence of database objects.

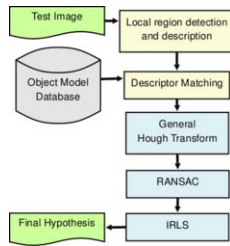


Figure 1. Block diagram of the Lowe object recognition method



Figure 2. Mobile robot platform used for the experiments.

3. Experimental Results

In this section we explain the experimental setup used for this paper and the results obtained in tests. Training images are close-ups of forty objects appearing in some panoramas, taken with a digital camera at a resolution of 1 megapixel. There is one training image per object, corresponding to its frontal view, except in some cases where the object can be found in two different views, i.e. fire extinguisher. This later case is treated as having two different objects. Descriptor vectors are extracted from training images and stored as a pre-processing step.

The testing dataset consists of five sequences of 11-22 panoramas taken with our mobile robot in our facilities. Panoramic images have a size about 6000x500 pixels. Figure 3 shows panoramas used to test our approach with the detected objects, and Figure 4 shows hypotheses obtained from the system, prior to ground truth verification.

Four experiments are done in order to test the importance of using RANSAC as hypothesis outlier filtering stage prior to IRLS, and also to test the performance loss and speedup of using approximate nearest neighbour over exhaustive search in matching stage. We also provide timing results of matching, Hough clustering and affine estimation stages.

As can be observed in Table 1, RANSAC provides no improvement over precision or recall. It provides better affine parameters initialization than the similarity transform obtained from Hough clustering step, so IRLS converges quicker than if RANSAC is not used. Although it has to be noted that neither performance nor speed are altered in a significant way, usually RANSAC filters more outliers than IRLS, resulting in more accurate pose estimation.

The processing time for each panorama, ignoring panorama acquisition time and feature extraction step, is about 1 second for matching using FLANN library set at 90% of retrieval precision, 5 ms for Hough Transform and 1-2 ms for the affine stage (RANSAC+IRLS or IRLS alone), so the processing can go near real-time. Tests have been done in a 3 Ghz Pentium IV with 1 Gb of RAM.

In order to test the applicability of object recognition in place classification task we set up the following test. Mean vectors are computed for each panorama sequence, except in the sequence we are testing, where one panorama is used to test and the rest are used to compute the mean vector, resulting in a leave-on-out evaluation strategy and a minimum distance classifier. The results of this test are shown in Table 2. As can be

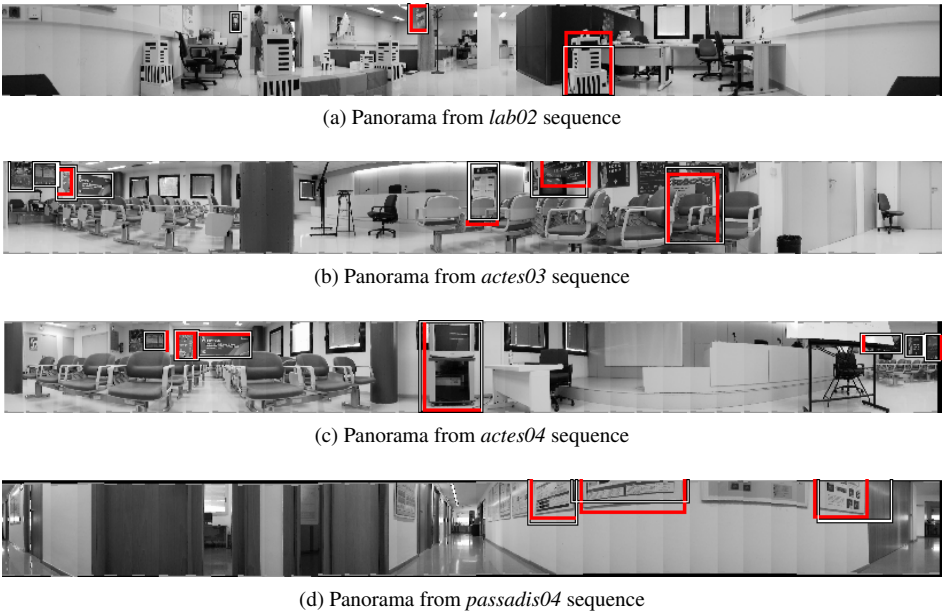


Figure 3. Final results obtained by our method in random panoramas of sequences. Only hypotheses consistent with ground truth are shown. Ground truth is shown in red.

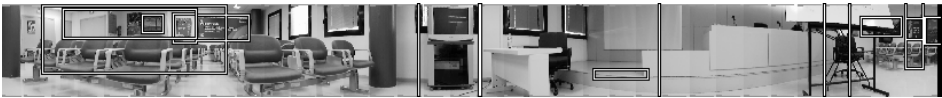


Figure 4. Object instances found in a random panorama.

| | ACC | PRE | REC |
|-------------------|------|------|------|
| RANSAC, no ANN | 0,81 | 0,37 | 0,53 |
| no RANSAC, no ANN | 0,81 | 0,37 | 0,53 |
| RANSAC, ANN | 0,81 | 0,36 | 0,53 |
| no RANSAC, ANN | 0,81 | 0,35 | 0,52 |

Table 1. Object recognition results.

| Sequence | Hit | Miss | Rate (%) |
|------------|-----|------|----------|
| lab02 | 10 | 1 | 91 |
| actes03 | 13 | 1 | 93 |
| actes04 | 11 | 3 | 79 |
| lab08 | 18 | 1 | 95 |
| passadis04 | 19 | 3 | 86 |

Table 2. Place recognition results.

seen, localization rate is good despite that object recognition only recalls 50% of objects and there is a significant amount of false positives.

4. Conclusions

In this work we have reviewed the different data sources that can be exploited for the object recognition task in the context of mobile robotics. Also, we tested a state-of-the-art method in images taken with a mobile robot to show one of the applications of object recognition in the robotics field. Although we have used a simple classification method

for place recognition, it has produced good results despite the modest performance of the object recognition method.

Furthermore, object recognition as a middle stage between image data and place recognition helps to reduce the complexity of the place descriptor while increasing its robustness. It also can help to reduce irrelevant information and noise from the background that could confuse other approaches to localization. For example in [13] the authors report false matches between individual features to be the most significant problem of the method. In this approach we only take into account *natural landmarks* formed with groups of locally coherent features, filtering potential false matches arising from spurious features. A possible extension to our approach would be incorporating position information to the detected objects along with a degree of mobility (for example a window would have mobility zero, while a chair would have a higher value).

It has to be noted that not all object recognition methods are suitable for mobile robotics environment, as robots need to take decisions in a relatively short time and imaging conditions and hardware used to take the pictures decrease the quality of input data.

In future work we plan to review the main state-of-the-art object recognition methods to find insight to develop a robust object recognition method that can be applied in mobile robotics.

Acknowledgements

This work has been partially funded by the FI grant and the BE grant from the AGAUR, the 2005/SGR/00093 project, supported by the Generalitat de Catalunya, the MIDCBR project grant TIN 200615140C0301, TIN 200615308C0202 and FEDER funds. The authors would also like to thank Marius Muja and David Lowe for making available the FLANN library.

References

- [1] S. Ekvall, P. Jensfelt, and D. Kragic. Integrating Active Mobile Robot Object Recognition and SLAM in Natural Environments. *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 5792–5797, 2006.
- [2] B. Leibe, A. Leonardis, and B. Schiele. Combined object categorization and segmentation with an implicit shape model. *Workshop on Statistical Learning in Computer Vision, ECCV*, pages 17–32, 2004.
- [3] D.G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [4] J. Matas, O. Chum, M. Urban, and T. Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image and Vision Computing*, 22(10):761–767, 2004.
- [5] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. *Proc. ECCV*, 1:128–142, 2002.
- [6] Marius Muja and D.G. Lowe. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. *Preprint*, 2008.
- [7] E. Murphy-Chutorian, S. Aboutalib, and J. Triesch. Analysis of a biologically-inspired system for real-time object recognition. *Cognitive Science Online*, 3(2):1–14, 2005.
- [8] D. Nister and H. Stewenius. Scalable recognition with a vocabulary tree. *Proc. CVPR*, pages 2161–2168, 2006.
- [9] A. Opelt, A. Pinz, M. Fussenegger, and P. Auer. Generic object recognition with boosting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(3):416–431, 2006.

- [10] A. Opelt, A. Pinz, and A. Zisserman. A boundary-fragment-model for object detection. *Proc. ECCV*, 2:575–588, 2006.
- [11] A. Opelt, A. Pinz, and A. Zisserman. Incremental learning of object detectors using a visual shape alphabet. *Proc. CVPR*, 1(3-10):4, 2006.
- [12] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in Context. *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8, 2007.
- [13] A. Ramisa, A. Tapus, R. Lopez de Mantaras, and R. Toledo. Mobile Robot Localization using Panoramic Vision and Combination of Local Feature Region Detectors. *Proc. IEEE International Conference on Robotics and Automation, Pasadena, California*, pages 538–543, 2008.
- [14] J. Sivic and A. Zisserman. Video Google: a text retrieval approach to object matching in videos. *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 1470–1477, 2003.
- [15] A. Torralba, KP Murphy, WT Freeman, and MA Rubin. Context-based vision system for place and object recognition. *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 273–280, 2003.
- [16] S. Vasudevan, S. Gächter, V. Nguyen, and R. Siegwart. Cognitive maps for mobile robots - an object based approach. *Robotics and Autonomous Systems*, 55(5):359–371, 2007.
- [17] X. Yu, L. Yi, C. Fermuller, and D. Doermann. Object Detection Using A Shape Codebook. *Unpublished*.

Motion Segmentation: a Review

Luca ZAPPELLA ^aXavier LLADÓ ^a and Joaquim SALVI ^a

^a *Institute of Informatics and Applications, University of Girona, Girona (Spain)*

Abstract. Motion segmentation is an essential process for many computer vision algorithms. During the last decade, a large amount of work has been trying to tackle this challenge, however, performances of most of them still fall far behind human perception. In this paper the motion segmentation problem is studied, analyzing and reviewing the most important and newest techniques. We propose a classification of all these techniques into different categories according to their main principle and features. Moreover, we point out their strengths and weaknesses and finally we suggest further research directions.

Keywords. Computer Vision, Motion Analysis, Motion Segmentation

Introduction

Motion segmentation aims at decomposing a video in moving objects and background. In many computer vision algorithms this decomposition is the first fundamental step. It is an essential building block for robotics, inspection, metrology, video surveillance, video indexing, traffic monitoring and many other applications. A great number of researchers has focused on the segmentation problem and this testifies the relevance of the topic. However, despite the vast literature, performances of most of the algorithms still fall far behind human perception.

In this paper we present a review on the main motion segmentation approaches with the aim of pointing out their strengths and weaknesses and suggesting new research directions. Our work is structured as follows. In the next section, a general overview of motion segmentation is presented. We describe common issues and the main attributes that should be considered when studying this type of algorithms. Furthermore, a classification among the different strategies is proposed. In section 2 the main ideas of each category are analyzed, reviewing also the most recent and important techniques. Finally, in section 3 general considerations are discussed for each category and conclusions are drawn.

1. Motion Segmentation: Main Features

In this section Common issues are described, a possible classification of motion segmentation algorithms is proposed, and a description of the main attributes that should be considered when studying this type of algorithms is presented.

In order to obtain an automatic motion segmentation algorithm that can work with real images there are several issues that need to be solved, particularly important are:

noise, missing data and lack of a priori knowledge. One of the main problem is the presence of noise. For some applications the noise level can become critical. For instance, in underwater imaging there are some specific sub-sea phenomena like water turbidity, marine snow, rapid light attenuation, strong reflections, back-scattering, non-uniform lighting and dynamic lighting that dramatically degrade the quality of the images [1,2,3]. Blurring is also a common issue especially when motion is involved [3]. Another common problem is caused by the fact that moving objects can create occlusions, or even worst, the whole object can disappear and reappear in the scene. Finally, it is important to take into account that not always it is possible to have prior knowledge about the objects or about the number of objects in the scene [3].

The main attributes of a motion segmentation algorithm can be summarized as follows.

- *Feature-based or Dense-based*: In feature-based methods, the objects are represented by a limited number of points like corners or salient points, whereas dense methods compute a pixel-wise motion [4].
- *Occlusions*: it is the ability to deal with occlusions.
- *Multiple objects*: it is the ability to deal with more than one object in the scene.
- *Spatial continuity*: it is the ability to exploit spatial continuity.
- *Temporary stopping*: it is the ability to deal with temporary stop of the objects.
- *Robustness*: it is the ability to deal with noisy images (in case of feature based methods it is the position of the point to be affected by noise but not the data association).
- *Sequentiality*: it is the ability to work incrementally, this means for example that the algorithm is able to exploit information that was not present at the beginning of the sequence.
- *Missing data*: it is the ability to deal with missing data.
- *Non-rigid object*: it is the ability to deal with non-rigid objects.
- *Camera model*: if it is required, which camera model is used (orthographic, para-perspective or perspective).

Furthermore, if the aim is to develop a generic algorithm able to deal in many unpredictable situations there are some algorithm features that may be considered as a drawback, specifically:

- *Prior knowledge*: any form of prior knowledge that may be required.
- *Training*: some algorithms require a training step.

Motion segmentation literature is wide. In order to make the overview easier to read and to create a bit of order, algorithms will be divided into categories which represent the main principle underlying the approach. The division is not meant to be tight, in fact some of the algorithms could be placed in more than one group. The categories identified are: *Image Difference*, *Statistical* (further divided into Maximum A posteriori Probability, Particle Filter and Expectation Maximization), *Optical Flow*, *Wavelets*, *Layers* and *Factorization*. For each category some articles, among the most representative and the newest proposals are analyzed. Table 1 offers a compact at-a-glance overview of the algorithms examined in this work with respect to the most important attributes of a motion segmentation algorithm.

2. Main Motion Segmentation Techniques

In this section we review the most important motion segmentation categories.

Image difference is one of the simplest and most used technique for detecting changes. It consists in thresholding the intensity difference of two frames pixel by pixel. The result is a coarse map of the temporal changes. An example of an image sequence and the image difference result is shown in figure 1. Despite its simplicity, this technique cannot be used in its basic version because it is really sensitive to noise. Moreover, when the camera is moving the whole image is changing and, if the frame rate is not high enough, the result would not provide any useful information. However, there are a few techniques based on this idea. The key point is to compute a rough map of the changing areas and for each blob to extract spatial or temporal information in order to track the region. Usually different strategies to make the algorithm more robust against noise and light changes are also used. Examples of this technique can be found in [1,5,6,7].

Statistic theory is widely used in the motion segmentation field. In fact, motion segmentation can be seen as a classification problem where each pixel has to be classified as background or foreground. Statistical approaches can be further divided depending on the framework used. Common frameworks are Maximum A posteriori Probability (MAP), Particle Filter (PF) and Expectation Maximization (EM). Statistical approaches provide a general tool that can be used in a very different way depending on the specific technique.

MAP is based on Bayes rule:

$$P(w_j|x) = \frac{p(x|w_j)P(w_j)}{\sum_{i=1}^c p(x|w_i)P(w_i)}$$

where x is the object to be classified (usually the pixel), $w_1..w_c$ are the c classes (usually background or foreground), $P(w_j|x)$ is the “a posteriori probability”, $p(x|w_j)$ is the conditional density, $P(w_j)$ is the “a priori probability” and $\sum_{i=1}^c p(x|w_i)P(w_i)$ is the “density function”. MAP classifies x as belonging to the class w which maximizes the “a posteriori probability”. MAP is often used in combination with other techniques. For example, in [8] is combined with a Probabilistic Data Association Filter. In [9] MAP is used together with level sets incorporating motion information. In [10] the MAP frame-

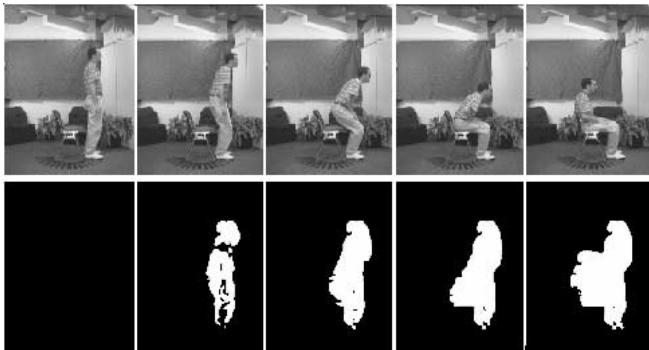


Figure 1. Example of image sequence and its image difference result. Sequence taken from [30].

work is used to combine and exploit the interdependence between motion estimation, segmentation and super resolution.

Another widely used statistical method is PF. The main aim of PF is to track the evolution of a variable over time. The basis of the method is to construct a sample-based representation of the probability density function. Basically, a series of actions are taken, each of them modifying the state of the variable according to some model. Multiple copies of the variable state (particles) are kept, each one with a weight that signifies the quality of that specific particle. An estimation of the variable can be computed as a weighted sum of all the particles. PF is an iterative algorithm, each iteration is composed by prediction and update. After each action the particles are modified according to the model (prediction), then each particle weight is re-evaluated according to the information extracted from an observation (update). At every iteration particles with small weights are eliminated [31]. An example of PF used in segmentation can be found in [11] where some well known algorithms for object segmentation using spatial information, such as geometric active contours and level sets, are unified within a PF framework.

EM is also a frequently exploited tool. The EM algorithm is an efficient iterative procedure to compute the Maximum Likelihood (ML) estimate in presence of missing or hidden data. ML consists in estimating the model parameter(s) that most likely represent the observed data. Each iteration of EM is composed by the E-step and the M-step. In the E-step the missing data are estimated using the conditional expectation, while in the M-step the likelihood function is maximized. Convergence is assured since the algorithm is guaranteed to increase the likelihood at each iteration [32]. As an example, in [3] an algorithm which combines EM and Extended-Markov Random Field is presented.

Another group of motion segmentation algorithms that we have identified is the one based on wavelets. These methods exploit the ability of wavelets to perform analysis of the different frequency components of the images, and then study each component with a resolution matched to its scale. Usually wavelet multi-scale decomposition is used in order to reduce the noise and in conjunction with other approaches, such as optical flow, applied at different scales. For instance, in [12] Wiskott combines Optical Flow with Gabor-wavelets in order to overcome the aperture problem. Furthermore, he extracts and tracks edges using the information provided by the Mallat-wavelets. Finally, the results are merged in order to obtain a robust segmentation. A different approach is presented



Figure 2. Example of Optical Flow: darker areas are the vectors of apparent velocities with length greater than zero

in [13] where the motion segmentation algorithm is based on Galilean wavelets. These wavelets behave as matched filters and perform minimum mean-squared error estimations of velocity, orientation, scale and spatio-temporal positions. This information is finally used for tracking and segmenting the objects.

Optical Flow (OF) is a vector motion field which describes the distribution of the apparent velocities of brightness patterns in a sequence, figure 2. Like image difference, OF is an old concept greatly exploited in computer vision. It was first formalized and computed for image sequences by Horn and Schunck in the 1980 [33], but the idea of using discontinuities in the OF in order to segment moving objects is even older. Since the work of Horn and Schunck, many other approaches have been proposed. In the past the main limitations of such methods were the high sensitivity to noise and the high computational cost. Nowadays, thanks to the high process speed of computers and to improvements made by research, OF is widely used. In [14] a method to segment multiple rigid-body motions using Line Optical Flow is presented.

The key idea of layers based techniques is to understand which are the different depth layers in the image and which objects (or which part of an articulated object) lie on which layer. This approach is often used in stereo vision as it is easier to compute the depth distance. However, without computing the depth it is possible to estimate which objects move on similar planes. This is extremely useful as it helps to solve the occlusion problem. In [4] a method for learning a layered representation of the scene is proposed. They initialize the method by first finding coarse moving components between every pair of frames. They divide the image in patches and find the rigid transformation that moved the patch from one frame to the next. The initial estimate is then refined using two minimization algorithms: $\alpha\beta$ -swap and α -expansion [34]. Figure 3 gives an example of how frame sequence can be used to learn the layers and the segments (objects or part of an objects) that lie on a specific layer.

Since *Tomasi and Kanade (1992)* [35] introduced a factorization technique to recover structure and motion using features tracked through a sequence of images, factorization methods have become very popular especially thanks to their simplicity. The idea is to factorize the trajectory matrix W (the matrix containing the position of the P features tracked throughout F frames) into two matrices: motion M and structure S , figure 4. If the origin of the world coordinate system is moved at the centroid of all the feature points, and in absence of noise, the trajectory matrix is at most rank 3. Exploiting this constraint, W can be decomposed and truncated using singular value decomposition

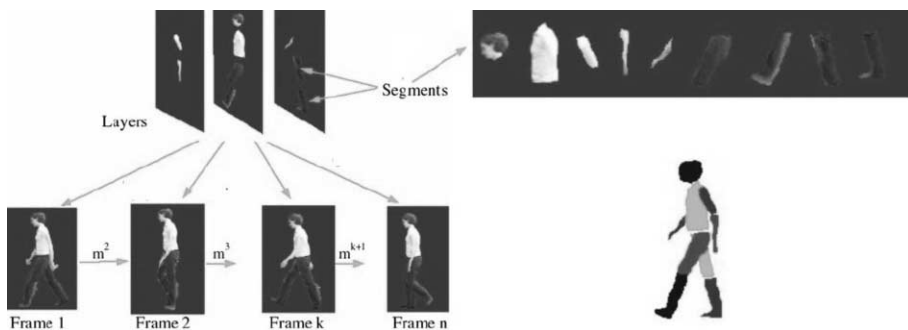


Figure 3. Example of Layers representation [4]

$$\begin{bmatrix} u_{11} & u_{12} & \dots & u_{1P} \\ v_{11} & v_{12} & \dots & v_{1P} \\ \vdots & \vdots & \ddots & \vdots \\ u_{F1} & u_{F2} & \dots & u_{FP} \\ v_{F1} & v_{F2} & \dots & v_{FP} \end{bmatrix} \sim \begin{bmatrix} t_{11}^1 & t_{12}^1 & t_{13}^1 \\ t_{21}^1 & t_{22}^1 & t_{23}^1 \\ \vdots & \vdots & \vdots \\ t_{11}^F & t_{12}^F & t_{13}^F \\ t_{21}^F & t_{22}^F & t_{23}^F \end{bmatrix} \begin{bmatrix} x_1 & y_1 & z_1 \\ x_2 & y_2 & z_2 \\ \vdots & \vdots & \vdots \\ x_P & y_P & z_P \end{bmatrix}$$

$W = 2F \times P \quad M = 2F \times 3 \quad S = 3 \times P$

Figure 4. Basic idea of structure from motion: factorize matrix W into M and S

(SVD), figure 5. Because the rows of the motion matrix are orthonormal, the matrix D

$$\begin{array}{cccc}
 W & U & D & V^T \\
 \begin{bmatrix} u_{11} & \dots & u_{1P} \\ v_{11} & \dots & v_{1P} \\ \vdots & \vdots & \vdots \\ u_{F1} & \dots & u_{FP} \\ v_{F1} & \dots & v_{FP} \end{bmatrix} & \overset{\text{SVD}}{=} & \begin{bmatrix} \text{column 1} \\ \text{column 2} \\ \text{column 3} \end{bmatrix} & \begin{bmatrix} \text{diag} \\ \text{diag} \\ \text{diag} \end{bmatrix} & \begin{bmatrix} \text{row 1} \\ \text{row 2} \\ \text{row 3} \end{bmatrix} \\
 2F \times P & & 2F \times r & r \times r & r \times P
 \end{array}$$

Figure 5. W can be decomposed and truncated using SVD and exploiting its rank deficiency

$$\begin{array}{cccc}
 U & D & V^T \\
 \begin{bmatrix} \text{column 1} \\ \text{column 2} \\ \text{column 3} \end{bmatrix} & \begin{bmatrix} \sqrt{D} \\ \sqrt{D} \\ \sqrt{D} \end{bmatrix} & \begin{bmatrix} \sqrt{D} \\ \sqrt{D} \\ \sqrt{D} \end{bmatrix} & \begin{bmatrix} \text{row 1} \\ \text{row 2} \\ \text{row 3} \end{bmatrix} \\
 2F \times 3 & 3 \times 3 & 3 \times 3 & 3 \times P \\
 \underbrace{\hspace{10em}}_M & & \underbrace{\hspace{10em}}_S &
 \end{array}$$

$W = M S$

Figure 6. Exploiting orthogonality constraints of the motion matrix, W can be decomposed into motion and structure, up to a scale factor

can be evaluated using orthogonality constraints, and finally decomposed, up to a scale factor, as shown in figure 6.

This initial algorithm works for one static object viewed from a moving camera which is modeled with the simplest affine camera model: the orthographic projection. Despite the fact that this method gives the 3D structure of the object and the motion of the camera, it has evident limits: it cannot really segment (it assumes that the features belong

to the same object), it can deal only with a single rigid object, it is very sensitive to noise and it is not able to deal with missing data and outliers. However, it was the first method of this family and the solution is mathematically elegant. From this initial structure from motion approach, and following the same idea of forcing the rank constraint, many approaches have been proposed in the field of motion segmentation. These methods are based on using the dimensionality of the subspace in which the image trajectories lie to perform the motion segmentation. For example, Costeira and Kanade proposed a factorization framework able to deal with multiple objects moving independently [15]. The assumption of independence between motion subspaces was also used in other approaches such as [16,17]. Moreover, Yan and Pollefeys [25] proposed a new general segmentation framework able to deal with different types of motion: independent, rigid, articulated and non-rigid motions. More recently, Julia et al. [29] extended Yan and Pollefeys approach to deal also with missing data in the image trajectories. Other remarkable factorization approaches are [19,21,22,24,26].

3. Conclusions

This review should have given an idea of how vast the motion segmentation literature is, and the fact that research in this field is still active (most of the papers presented were published after 2005) is a sign of the importance of the solution to this problem. On the other hand, effervescent research activity signifies that an outstanding solution has yet to be found.

As can be seen from Table 1, image difference is mainly based on dense representation of the objects. It combines simplicity and good overall results being able to deal with occlusions, multiple objects and non-rigid objects. The main problems of these techniques are the difficulty to deal with temporary stopping and with moving cameras. In order to be successful in these situations a history model of the background needs to be built. Furthermore, these algorithms are still very sensitive to noise and light changes.

Statistical approaches also use mainly dense based representation. These methods work well with multiple objects and are able to deal with occlusions and temporary stopping. In general they are robust as long as the model reflects the actual situation but they degrade quickly as the model fails to represent the reality. Finally, most of the statistic approaches require some kind of a priori knowledge.

Wavelets solutions seem to provide good results, wavelets were in fashion during the 90s and now it seems that the research interest is decreased. Their ability in performing multi resolution analysis could be exploited in order to extract information about the different depth planes in the scene, thus helping to solve the occlusion problem.

Optical flow is theoretically a good clue in order to segment motion. However, OF alone it is not enough since it cannot help to solve occlusions and temporal stopping. Moreover, these methods are sensitive to noise and light changes.

The layer solution is very interesting. It is probably the more natural solution for the occlusions. Note that human beings also use depth concept to solve this problem. The main drawback of this strategy is the level of complexity of the algorithm and the high number of parameters that need to be tuned.

Factorization methods are an elegant solution based on feature points. Besides, they provide not only the segmentation but they can be naturally connected to a structure from

motion algorithm in order to recover the 3D structure of the objects and the motion of the camera. Furthermore, they do not have any problem with temporary stopping because the features can be tracked even if the object is not moving (provided that this is a temporary situation). With respect to other approaches, factorization methods are particularly weak in terms of ability to deal with noise and outliers, and they have more problems to deal with non rigid objects because the non rigidity has to be explicitly taken into account. A quick glance at the table may catch the attention on two elements. The first is that, to the best of our knowledge, there is no factorization algorithm which is able to provide segmentation in an incrementally way. A technique able to provide a good object segmentation exploiting only few frames, and refining the solution with the time, it would be an important step ahead. The second is that with the exception of [26], spatial continuity is not exploited. These may suggest that the usage of this information may help to improve factorization method performances, especially in terms of robustness and ability to deal with occlusions.

Considering the aspects emerged in this review, we believe that a factorization approach could be a good starting point for proposing a novel segmentation technique. They are based on a powerful mathematical framework and they can provided easily segmentation and structure of the objects. In order to obtain more robust results it would be interesting to study different ways to merge factorization with spatial information and to exploit also the ability of statistical frameworks to find and use hidden information. These are the directions that we intend to follow in the near future.

References

- [1] A. Cavallaro, O. Steiger, and T. Ebrahimi, "Tracking Video Objects in Cluttered Background," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 4, pp. 575–584, 2005.
- [2] E. Trucco and K. Plakas, "Video tracking: A concise survey," *IEEE Journal of Oceanic Engineering*, vol. 31, no. 2, pp. 520–529, 2006.
- [3] R. Stolkin, A. Greig, M. Hodgetts, and J. Gilby, "An em/e-mrf algorithm for adaptive model based tracking in extremely poor visibility," *Image and Vision Computing*, vol. 26, no. 4, pp. 480–495, 2008.
- [4] M. P. Kumar, P. H. Torr, and A. Zisserman, "Learning layered motion segmentations of video," *International Journal of Computer Vision*, vol. 76, no. 3, pp. 301–319, 2008.
- [5] F.-H. Cheng and Y.-L. Chen, "Real time multiple objects tracking and identification based on discrete wavelet transform," *Pattern Recognition*, vol. 39, no. 6, pp. 1126–1139, 2006.
- [6] R. Li, S. Yu, and X. Yang, "Efficient spatio-temporal segmentation for extracting moving objects in video sequences," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 3, pp. 1161–1167, Aug. 2007.
- [7] A. Colombari, A. Fusiello, and V. Murino, "Segmentation and tracking of multiple video objects," *Pattern Recognition*, vol. 40, no. 4, pp. 1307–1317, 2007.
- [8] C. Rasmussen and G. D. Hager, "Probabilistic data association methods for tracking complex visual objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 560–576, 2001.
- [9] D. Cremers and S. Soatto, "Motion competition: A variational approach to piecewise parametric motion segmentation," *International Journal of Computer Vision*, vol. 62, no. 3, pp. 249–265, May 2005.
- [10] H. Shen, L. Zhang, B. Huang, and P. Li, "A map approach for joint motion estimation, segmentation, and super resolution," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 479–490, 2007.
- [11] N. Vaswani, A. Tannenbaum, and A. Yezzi, "Tracking deforming objects using particle filtering for geometric active contours," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 8, pp. 1470–1475, 2007.
- [12] L. Wiskott, "Segmentation from motion: Combining Gabor- and Mallat-wavelets to overcome aperture and correspondence problem," in *Proceedings of the 7th International Conference on Computer Analysis*

of *Images and Patterns*, G. Sommer, K. Daniilidis, and J. Pauli, Eds., vol. 1296. Heidelberg: Springer-Verlag, 1997, pp. 329–336.

- [13] M. Kong, J.-P. Leduc, B. Ghosh, and V. Wickerhauser, "Spatio-temporal continuous wavelet transforms for motion-based segmentation in real image sequences," *Proceedings of the International Conference on Image Processing*, vol. 2, pp. 662–666 vol.2, 4-7 Oct 1998.
- [14] J. Zhang, F. Shi, J. Wang, and Y. Liu, "3d motion segmentation from straight-line optical flow," in *Multimedia Content Analysis and Mining*, 2007, pp. 85–94.
- [15] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," *International Journal of Computer Vision*, vol. 29, no. 3, pp. 159–179, 1998.
- [16] N. Ichimura and F. Tomita, "Motion segmentation based on feature selection from shape matrix," *Systems and Computers in Japan*, vol. 31, no. 4, pp. 32–42, 2000.
- [17] K. Kanatani and C. Matsunaga, "Estimating the number of independent motions for multibody motion segmentation," in *Proceedings of the Fifth Asian Conference on Computer Vision*, vol. 1, Jan 2002, pp. 7–12.
- [18] M. Brand, "Incremental singular value decomposition of uncertain data with missing values," in *European Conference on Computer Vision*, 2002, pp. 707–720.
- [19] L. Zelnik-Manor and M. Irani, "Degeneracies, dependencies and their implications in multi-body and multi-sequence factorizations," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. II–287–93 vol.2, 18-20 June 2003.
- [20] H. Zhou and T. S. Huang, "Recovering articulated motion with a hierarchical factorization method," in *Gesture Workshop*, 2003, pp. 140–151.
- [21] Y. Sugaya and K. Kanatani, "Geometric structure of degeneracy for multi-body motion segmentation," in *Statistical Methods in Video Processing*, 2004, pp. 13–25.
- [22] L. Zelnik-Manor and M. Irani, "Temporal factorization vs. spatial factorization," in *European Conference on Computer Vision*, vol. 2. Springer Berlin / Heidelberg, 2004, pp. 434–445.
- [23] A. Gruber and Y. Weiss, "Multibody factorization with uncertainty and missing data using the em algorithm," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. I–707–I–714 Vol.1, 27 June-2 July 2004.
- [24] R. Vidal and R. Hartley, "Motion segmentation with missing data using powerfactorization and gpca," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. II–310–II–316 Vol.2, 27 June-2 July 2004.
- [25] J. Yan and M. Pollefeys, "A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate," in *European Conference on Computer Vision*, 2006, pp. IV: 94–106.
- [26] A. Gruber and Y. Weiss, "Incorporating non-motion cues into 3d motion segmentation," in *European Conference on Computer Vision*, 2006, pp. 84–97.
- [27] X. Llado, A. D. Bue, and L. Agapito, "Euclidean reconstruction of deformable structure using a perspective camera with varying intrinsic parameters," *18th International Conference on Pattern Recognition*, vol. 1, pp. 139–142, 2006.
- [28] A. Goh and R. Vidal, "Segmenting motions of different types by unsupervised manifold clustering," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–6, 17-22 June 2007.
- [29] C. Julià, A. Sappa, F. Lumberras, J. Serrat, and A. Lopez, "Motion segmentation from feature trajectories with missing data," in *Iberian Conference on Pattern Recognition and Image Analysis*, 2007, pp. I: 483–490.
- [30] A. Bobick and J. Davis, "An appearance-based representation of action," *IEEE International Conference on Pattern Recognition*, pp. 307–312, 1996.
- [31] I. Rekleitis, "Cooperative localization and multi-robot exploration," PhD in Computer Science, School of Computer Science, McGill University, Montreal, Quebec, Canada, 2003.
- [32] S. Borman, "The expectation maximization algorithm – a short tutorial," Jul. 2004.
- [33] B. K. Horn and B. G. Schunck, "Determining optical flow," Cambridge, MA, USA, Tech. Rep., 1980.
- [34] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," in *International Conference on Computer Vision*, 1999, pp. 377–384.
- [35] C. Tomasi and T. Kanade, "Shape and motion from image streams under orthography: a factorization method," *International Journal of Computer Vision*, vol. 9, no. 2, pp. 137–154, 1992.

This page intentionally left blank

Uncertainty in AI

This page intentionally left blank

On Fuzzy Description Logics

Àngel GARCÍA-CERDAÑA, Francesc ESTEVA
IIIA, Artificial Intelligence Research Institute
CSIC, Spanish Council for Scientific Research
Campus UAB, 08193 Bellaterra, Spain
e-mail: {angel, esteva}@iiia.csic.es

Abstract.

Description Logics (DLs) are knowledge representation languages, particularly suited to specify formal ontologies. DLs have been studied extensively over the last two decades. Fuzzy Description Logics (FDLs) incorporate vague concepts modeling them as fuzzy sets. Following ideas from Hájek, we propose the use of truth constants in the languages of description. In particular we introduce the languages $\mathcal{ALC}_{L^*}(\mathbf{s})$ and $\mathcal{ALC}_{L^*}(\mathbf{s})$ as an adequate syntactical counterpart of some semantic calculi given in different works dealing with FDLs. In addition we give completeness results for some languages \mathcal{ALC} -like.

Keywords. Description Logics, t-Norm Based Fuzzy Logics without and with truth constants, Fuzzy Description Logics

1. Description Logics, Fuzzy Logics, and Fuzzy Description Logics

Methods and tools suited to give a high-level description of the world and to implement intelligent systems play a key role in the area of knowledge representation and reasoning. The systems developed in the 70's were mainly of two kinds: ones based on first-order classical logic, and another ones (like frames, semantic networks, etc.) that take into account some practical approaches to structure the knowledge without using any particular logic. Description Logics (DLs) are knowledge representation languages (particularly suited to specify formal ontologies), which have been studied extensively over the last two decades. A full reference manual of the field is [1]. It is important to stress that in the last decade DLs have improved their popularity among scientists in Computer Science and Artificial Intelligence as a consequence of their implementation inside the Semantic Web (SW). Ontologies play a key role in the SW, and ontologies based on DLs have been very successful in this context. DLs have an important role because they are the theoretical counterpart of the Web Ontology Language OWL DL.

DLs are a family of formalisms describing a domain through a knowledge base (KB) containing the definition of relevant domain concepts (called TBox) and a specification of properties of the domain instances (called ABox). The vocabulary of DLs consists of *concepts*, which denote sets of individuals, and *roles*, which denote binary relations among individuals. From atomic concepts and roles DL systems allow, by means of *constructors*, to build complex descriptions of both concepts and roles. One of the main

issues of DLs is the fact that the semantics is given in a Tarski-style presentation and the statements in the TBox and in the ABox can be identified with formulae in first-order logic or a slight extension of it, and hence we can use reasoning to obtain implicit knowledge from the explicit knowledge in the KB.

Obviously, in real applications the knowledge used is usually imperfect and has to address situations of uncertainty, imprecision and vagueness. From a real world viewpoint, vague concepts like “patient with a high fever” and “person living near Paris” have to be considered. These concepts and relations (that include “vagueness”) have suggested to model DL concepts as fuzzy sets. With the exception of an isolated paper of Yen [23] published in 1991, it is at the end of the last decade (from 1998) when several proposals of FDLs (Fuzzy Description Logics) were introduced like, for instance, the ones by Tresp and Molitor [22] and Straccia (e.g. [18]). However, the fuzzy framework behind this initial works is very limited and uses a “minimalistic” apparatus of fuzzy logic. Namely these papers are logically based on a multivalued calculus using the operations “max”, “min” and the involutive negation $\neg x = 1 - x$ (for more information see [11]).

On the other hand, the growing in the last decade on the field of formal systems of Fuzzy Logics provides to FDLs with a powerful theoretical machinery. The development of that field is intimately linked to the book “Metamathematics of Fuzzy Logics” by Petr Hájek [9], published in 1998. Hájek shows the connection of fuzzy logic systems with many-valued residuated lattices based on continuous triangular norms (*t*-norms). He proposes a formal system called Basic fuzzy Logic (*BL*) and conjectures that it is sound and complete with respect to the structures defined in the unit real interval $[0, 1]$ by continuous *t*-norms and their residua.¹

In the paper *Making fuzzy logic description more general* [10], published in 2005, Hájek proposes to go beyond the minimalistic approach and to deal with FDL taking as basis *t*-norm based fuzzy logics with the aim of enriching the expressive possibilities in FDL (see also [11]). This change of view gives a wide number of choices on which a DL can be based: for every particular problem we can consider the fuzzy logic that seems to be more adequate. As an example, Hájek studies the FDL associated with the language \mathcal{ALC} . After this work, researchers on FDLs have developed approaches based on the spirit of Hájek’s paper, even though their work is more related to expressiveness and algorithms (see for instance [20,17,21]).

One of the main motivations of the present work is based on the following consideration: since the axioms of the bases of knowledge in FDLs include truth degrees (see for instance [18]), a natural choice is to include symbols for these degrees in both, the description language and the fuzzy logic where that language is interpreted. In the recent years the topic of fuzzy logics with truth constant in the language has received a renewed impulse. Relevant works concerning to such formal systems are the first papers of Pavelka [13,14,15] and more recent works such as [9,6,16,3]. In this paper we propose two new families of description languages, denoted by $\mathcal{ALC}_{L^*}(S)$ and $\mathcal{ALC}_{L_{\sim}}(S)$. This languages are extensions of the language \mathcal{ALC} considered by Hájek in [10]. We define their semantics and describe the corresponding bases of knowledge (TBox and ABox) from a syntactic and semantic perspective.

¹The conjecture was proven in [2]. Later, Esteva and Godo [4] introduced the logic *MTL*, a weakening of *BL*, which propositional fragment is proved to be the logic of left continuous *t*-norms and their residuum [12].

| $*$ | Minimum (Gödel) | Product (of real numbers) | Łukasiewicz |
|---------------------|---|---|----------------------|
| $x * y$ | $\min(x, y)$ | $x \cdot y$ | $\max(0, x + y - 1)$ |
| $x \rightarrow_* y$ | $\begin{cases} 1, & \text{if } x \leq y \\ y, & \text{otherwise} \end{cases}$ | $\begin{cases} 1, & \text{if } x \leq y \\ y/x, & \text{otherwise} \end{cases}$ | $\min(1, 1 - x + y)$ |
| n_* | $\begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases}$ | $\begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise} \end{cases}$ | $1 - x$ |

Table 1. The three main continuous t -norms.

Next section summarizes some basic notions of propositional and first order t -norm based (fuzzy) logics without and with truth constants. In Section 3 we describe two new families of fuzzy description languages with truth constants, denoted by $\mathcal{ALC}_{L^*(S)}$ and $\mathcal{ALC}_{L^*(S)}$, and the corresponding semantics. Finally, in Section 4 we present some ideas to find the fuzzy logics adequate as a basis for FDLs in general, and for the languages $\mathcal{ALC}_{L^*(S)}$ and $\mathcal{ALC}_{L^*(S)}$ in particular.

2. Fuzzy Logic: basics

A *triangular norm* (t -norm for short) is a binary operation defined on the real interval $[0, 1]$ satisfying the following properties: it is associative and commutative, non decreasing in both arguments and having 1 as unit element. If $*$ is a continuous t -norm, then there exists an unique operation \rightarrow_* satisfying, for all $a, b, c \in [0, 1]$, the condition $a * c \leq b \Leftrightarrow c \leq a \rightarrow_* b$. This operation is called the *residuum* of the t -norm. The *negation* n_* associated with $*$ is defined as follows: $n_*(x) = x \rightarrow_* 0$. Table 1 shows the main t -norms (Gödel, Product and Łukasiewicz) with their residua and the associated negations. Any continuous t -norm can be expressed as an ordinal sum of copies of these three t -norms. A *triangular conorm* (t -conorm for short), is a binary operation defined on $[0, 1]$ associative, commutative, non decreasing in both arguments, and having 0 as unit element. Table 2 shows the dual t -conorms of the three basic continuous t -norms.

| \oplus | Maximum (Gödel) | Sum | Łukasiewicz |
|--------------|-----------------|---------------------|------------------|
| $x \oplus y$ | $\max(x, y)$ | $x + y - x \cdot y$ | $\min(1, x + y)$ |

Table 2. The dual t -conorms corresponding to the three main continuous t -norms.

Fuzzy propositional logics. **BL** (Basic fuzzy Logic) is the propositional logic in the language $\langle \&, \rightarrow, \bar{0} \rangle$ defined by the inference rule of *Modus Ponens* and the following schemata (taking \rightarrow as the least binding connective):

- (BL1) $(\varphi \rightarrow \psi) \rightarrow ((\psi \rightarrow \chi) \rightarrow (\varphi \rightarrow \chi))$
- (BL2) $\varphi \& \psi \rightarrow \varphi$
- (BL3) $\varphi \& \psi \rightarrow \psi \& \varphi$
- (BL4) $\varphi \& (\varphi \rightarrow \psi) \rightarrow \psi \& (\psi \rightarrow \varphi)$
- (BL5a) $(\varphi \rightarrow (\psi \rightarrow \chi)) \rightarrow (\varphi \& \psi \rightarrow \chi)$
- (BL5b) $(\varphi \& \psi \rightarrow \chi) \rightarrow (\varphi \rightarrow (\psi \rightarrow \chi))$
- (BL6) $((\varphi \rightarrow \psi) \rightarrow \chi) \rightarrow (((\psi \rightarrow \varphi) \rightarrow \chi) \rightarrow \chi)$
- (BL7) $\bar{0} \rightarrow \varphi$

The usual defined connectives are introduced as follows:

$$\begin{aligned}\varphi \vee \psi &:= ((\varphi \rightarrow \psi) \rightarrow \psi) \wedge ((\psi \rightarrow \varphi) \rightarrow \varphi), & \varphi \wedge \psi &:= \varphi \& (\varphi \rightarrow \psi), \\ \varphi \leftrightarrow \psi &:= (\varphi \rightarrow \psi) \& (\psi \rightarrow \varphi), & \neg \varphi &:= \varphi \rightarrow \bar{0}, & \bar{1} &:= \neg \bar{0}.\end{aligned}$$

Łukasiewicz, Product and Gödel Logics can be obtained as axiomatic extensions of BL with the following axioms: $\neg\neg\varphi \rightarrow \varphi$ for Łukasiewicz, $(\varphi \wedge \neg\varphi) \rightarrow \bar{0}$, and $\neg\neg\chi \rightarrow (((\varphi \& \chi) \rightarrow (\psi \& \chi)) \rightarrow (\varphi \rightarrow \psi))$ for Product, and $\varphi \rightarrow \varphi \& \varphi$ for Gödel.

An *evaluation of propositional variables* is a mapping e assigning to each variable p a *true value* $e(p) \in [0, 1]$. Given a continuous t -norm $*$, the evaluation e is extended inductively to a mapping of all formulas into the so-called *standard algebra* $[0, 1]_* = \langle [0, 1], *, \rightarrow_*, \max, \min, 0, 1 \rangle$ defined on $[0, 1]$ by the t -norm and its residuum in the following way: $e(\varphi \& \psi) = e(\varphi) * e(\psi)$; $e(\varphi \rightarrow \psi) = e(\varphi) \rightarrow_* e(\psi)$; $e(\bar{0}) = 0$. It is well known that the formal system BL is sound and complete w.r.t. the structures defined in $[0, 1]$ by continuous t -norms and their residua, i.e., for every formula φ , every continuous t -norm $*$, an every evaluation e on $[0, 1]$, φ is derivable in BL iff $e(\varphi) = 1$ (see [2]). In this sense, it is also well known that Łukasiewicz (Product, Gödel) Logic is the logic of the t -norm of Łukasiewicz (Product, Gödel).

In [6] the logic of each continuous t -norm $*$ and its residuum, denoted by L^* , is proved to be a finite axiomatic extension of BL . An interesting fact is that L^* is complete with respect to evaluations over the standard algebra $[0, 1]_*$. When the defined negation in L^* is not involutive, a new logic expanding can be considered. This logic, denoted by L^*_\sim , is obtained by adding an involutive negation that allows: a) the definition by duality of a multiplicative disjunction by $\varphi \vee \psi := \sim (\sim \varphi \& \sim \psi)$, and b) the usual generalization used in fuzzy logic for the classical implication, i.e., $\varphi \hookrightarrow \psi := \sim \varphi \vee \psi$. The logics L^*_\sim can be presented in the language $\langle \&, \rightarrow, \Delta, \sim, \bar{0} \rangle$, where Δ (the Baaz Delta) and \sim (a involutive negation) are unary connectives. In the axiomatic extensions of BL with $\varphi \wedge \neg\varphi \rightarrow \bar{0}$ as a theorem, the connective Δ is definable and so the corresponding logic L^*_\sim can be presented without the connective Δ in the language. For more information of these logics, their axiomatizations and completeness theorems see [5, 8].

Fuzzy predicate logics. The language of the *basic fuzzy predicate logic* $BL\forall$ consists of a set of *predicate symbols* $\mathcal{P} = \{P, Q, \dots\}$, each together with its *arity* $n \geq 1$ and a set of *constant symbols* $\mathcal{C} = \{c, d, \dots\}$. The logical symbols are *variable symbols* x, y, \dots , *connectives* $\&, \rightarrow, \bar{0}$ and *quantifiers* \forall, \exists . Other connectives $(\vee, \wedge, \neg, \leftrightarrow, \bar{1})$ are defined as in BL . *Terms* are constant symbols and variable symbols. An *atomic formula* is an expression of the form $P(t_1, \dots, t_n)$, where P is a predicate letter of arity n and t_1, \dots, t_n are terms. The set of predicate formulas of $BL\forall$ is the smallest set satisfying the following conditions:

- each atomic formula is a formula,
- $\bar{0}$ is a formula,
- if φ and ψ are formulas, then $\varphi \& \psi$ and $\varphi \rightarrow \psi$ are formulas,
- if φ is a formula, and x is a variable, then $(\forall x)\varphi$ and $(\exists x)\varphi$ are formulas.

A *fuzzy interpretation* for our language is a tuple $\mathbf{M} = \langle M, (r_P)_{P \in \mathcal{P}}, (m_c)_{c \in \mathcal{C}} \rangle$, where M is a set, for each n -ary predicate symbol P , r_P is a fuzzy n -ary relation $M^n \rightarrow [0, 1]$; and for each constant symbol c , m_c is an element of M .

Given a continuous t -norm $*$, an \mathbf{M} -evaluation of the variables assigns to each variable x an element $v(x)$ of M . From M and v we define the *true value of a term* t in the following way: $\|t\|_{\mathbf{M},v} = v(t)$ when t is a variable, and $\|t\|_{\mathbf{M},v} = m_c$ when t is a constant c . The *truth value of a formula* φ for an evaluation v , denoted by $\|\varphi\|_{\mathbf{M},v}^*$, is a value in $[0, 1]$ defined inductively as follows:

- $r_P(\|t_1\|_{\mathbf{M},v}, \dots, \|t_n\|_{\mathbf{M},v})$, if $\varphi = P(t_1, \dots, t_n)$,
- 0, if $\varphi = \bar{0}$,
- $\|\alpha\|_{\mathbf{M},v}^* * \|\beta\|_{\mathbf{M},v}^*$, if $\varphi = \alpha \& \beta$,
- $\|\alpha\|_{\mathbf{M},v}^* \rightarrow_* \|\beta\|_{\mathbf{M},v}^*$, if $\varphi = \alpha \rightarrow \beta$,
- $\inf\{\|\varphi\|_{\mathbf{M},v[x/a]}^* : a \in M\}$, if $\varphi = (\forall x)\alpha$,
- $\sup\{\|\varphi\|_{\mathbf{M},v[x/a]}^* : a \in M\}$, if $\varphi = (\exists x)\alpha$,

where $v[x/a]$ is a \mathbf{M} -evaluation taking the same values as v for $x \neq a$.

If M is finite, the infimum (supremum) coincides with the minimum (maximum). We define the *truth value of a formula* φ as $\|\varphi\|_{\mathbf{M}}^* := \inf\{\|\varphi\|_{\mathbf{M},v}^* : v \text{ is an } \mathbf{M}\text{-evaluation}\}$. A formula φ is a **-tautology* if $\|\varphi\|_{\mathbf{M}}^* = 1$ for every fuzzy interpretation \mathbf{M} ; φ is a *standard tautology* if it is a **-tautology* for each continuous t -norm $*$. The following standard tautologies are taken as axioms of the basic fuzzy predicate logic $BL\forall$ [9]:

- 1) the axioms of BL , and
- 2) the following axioms on quantifiers:
 - ($\forall 1$) $(\forall x)\varphi(x) \rightarrow \varphi(t)$ (t substitutable for x in $\varphi(x)$),
 - ($\exists 1$) $\varphi(t) \rightarrow (\exists x)\varphi(x)$ (t substitutable for x in $\varphi(x)$),
 - ($\forall 2$) $(\forall x)(\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow (\forall x)\psi)$ (x not free in φ),
 - ($\exists 2$) $(\forall x)(\varphi \rightarrow \psi) \rightarrow ((\exists x)\varphi \rightarrow \psi)$ (x not free in ψ),
 - ($\forall 3$) $(\forall x)(\varphi \vee \psi) \rightarrow (\forall x)\varphi \vee \psi$ (x not free in ψ).

Deduction rules are (as in classical logic) *Modus Ponens* and *Generalization*. Given this, the notions of *proof*, *provability*, *theory*, etc., are defined in the usual way. Let \mathcal{C} be an axiomatic extension of BL . $\mathcal{C}\forall$ is obtained by taking the axioms and rules of $BL\forall$ plus the axioms characterizing \mathcal{C} . Thus, given a continuous t -norm $*$, the predicate logic $L^*\forall$ is the logic obtained from L^* by adding to its axiomatization the schemas for quantifiers and the rule of generalization.

2.1. Adding truth constants to the language

T -norm based fuzzy logics are a particular case of multi-valued logics. This is because truth values go from false (0) to true (1) and all the intermediate values represent the different degrees of truth, i.e., the *partial truth* of a formula. These logics can also be seen as logics of *comparative truth*. In fact, the residuum \rightarrow_* of a continuous t -norm $*$ satisfies the condition $x \rightarrow_* y = 1$ iff, $x \leq y$ for all $x, y \in [0, 1]$. This means that a formula $\varphi \rightarrow \psi$ is a logical consequence of a theory if the truth degree of φ is at most as high as the truth degree of ψ in any interpretation which is a model of the theory. However, the advantage of being a multi-valued logic is not used in current approaches since the semantic deduction of formulas do not take into account truth degrees. An elegant way to take benefit from the multi-valued approach is to introduce truth constants

into the language, as it is done by Pavelka in [13,14,15] and more recently in [9,6,16,3]. In fact the approach considered in the present paper is also based in this idea.

Given a continuous t-norm $*$, its residuum \rightarrow_* and its corresponding logic L^* , let $\mathbf{S} = \langle S, *, \rightarrow_*, \max, \min, 0, 1 \rangle$ be a *countable* (i.e., finite or enumerable) subset closed under the operations of the corresponding standard algebra (a *subalgebra* of $[0, 1]_*$). The expansion of L^* adding into the language a truth constant \bar{r} for each $r \in S$, denoted by $L^*(\mathbf{S})$, is defined as follows:

- i) the language of $L^*(\mathbf{S})$ is the one of L^* plus a truth constant \bar{r} for each $r \in S$,
- ii) the axioms and rules of $L^*(\mathbf{S})$ are those of L^* plus the *book-keeping* axioms, that is, for each $r, s \in S \setminus \{0, 1\}$, the axioms $\bar{r} \& \bar{s} \leftrightarrow \overline{r * s}$ and $\bar{r} \rightarrow \bar{s} \leftrightarrow \overline{r \rightarrow_* s}$.

When the negation associated to the continuous t-norm $*$ is not involutive, the logic $L^*_\sim(\mathbf{S})$ can be defined in a similar way although in this case \mathbf{S} is a countable subalgebra of the algebra obtained by adding the operation of negation $\sim x := 1 - x$ to the operations of $[0, 1]_*$. The predicate logics $L^*\forall(\mathbf{S})$ and $L^*_\sim\forall(\mathbf{S})$ are respectively obtained from $L^*\forall$ and $L^*_\sim\forall$ by expanding the language with a truth constant \bar{r} for every $r \in S$ and by adding the book keeping axioms. The truth value of the formula \bar{r} is given by $\|\bar{r}\|_{\mathbf{M}}^* = r$. The logics $L^*\forall(\mathbf{S})$ and $L^*_\sim\forall(\mathbf{S})$ will be the counterpart of the description languages $\mathcal{ALC}_{L^*(\mathbf{S})}$ and $\mathcal{ALC}_{L^*_\sim(\mathbf{S})}$ presented in the next section.

3. The fuzzy description logics $\mathcal{ALC}_{L^*(\mathbf{S})}$ and $\mathcal{ALC}_{L^*_\sim(\mathbf{S})}$

The so-called *evaluated formulas*, i.e., formulas of type $\bar{r} \rightarrow \varphi$, where φ is a formula without truth constants are an interesting fragment of the fuzzy logics with truth constants in the language. This is the type of formulas used for the knowledge bases in recent papers about FDLs (cf.[18,19,21]) and it is the logic fragment underlying the languages $\mathcal{ALC}_{L^*(\mathbf{S})}$ and $\mathcal{ALC}_{L^*_\sim(\mathbf{S})}$ that we define in this section from a syntactic and semantic perspective. Then we introduce the notions of *TBox* (Terminological Box) and of *ABox* (Assertional Box) for that languages.

Syntax. In the languages of description we start from *atomic concepts* and *atomic roles*. Complex descriptions are built inductively with constructors of concepts. We will use the letter A for atomic concepts, the letter R for atomic roles and the letters C and D for descriptions of concepts. Given a continuous t-norm $*$ and a countable subalgebra \mathbf{S} of the corresponding standard algebra $[0, 1]_*$, let us consider the logic $L^*(\mathbf{S})$. Using the connectives $\bar{0}, \&, \rightarrow$, plus the truth constants $\{\bar{r} : r \in S\}$, and the quantifiers \forall, \exists , the descriptions of concepts in $\mathcal{ALC}_{L^*(\mathbf{S})}$ are built using the following syntactic rules:

| | | | |
|-------------------------|-------------------|-----------------------|------------------------------|
| $C, D \rightsquigarrow$ | A | | (atomic concept) |
| | $\bar{0}$ | | (falsum) |
| | \bar{r} | , for every $r \in S$ | (truth degrees) |
| | $C \& D$ | | (fusion) |
| | $C \rightarrow D$ | | (implication) |
| | $\forall R.C$ | | (universal quantification) |
| | $\exists R.C$ | | (existential quantification) |

Similarly, the language $\mathcal{ALC}_{L^*}(S)$ is defined adding to $\mathcal{ALC}_{L^*}(S)$ the connectives \sim and Δ . The syntactic rules for these connectives are the following:

$$\begin{array}{l} C \rightsquigarrow \sim C \mid \text{(involutive negation)} \\ \Delta C \mid \text{(Delta)} \end{array}$$

In both languages the connectives $\vee, \wedge, \neg, \bar{\cdot}$ are defined as follows:

$$\begin{aligned} C \wedge D &:= C \& (C \rightarrow D), \quad C \vee D := ((C \rightarrow D) \rightarrow D) \wedge ((D \rightarrow C) \rightarrow C), \\ \neg C &:= C \rightarrow \bar{0}, \quad \bar{\bar{1}} := \neg \bar{0}. \end{aligned}$$

The notion of *instance of a concept* allows us to read the formulas of both languages as formulas of corresponding the predicate fuzzy logic: for each term t (variable or constant), the *instance* $C(t)$ of a concept is defined as follows:

- $A(t)$ is the atomic formula in which A is interpreted as a unary predicate,
- $\bar{0}(t)$ is $\bar{0}$; and $\bar{r}(t)$ is \bar{r} ,
- $(C \& D)(t)$ and $(C \rightarrow D)(t)$ are $C(t) \& D(t)$ and $C(t) \rightarrow D(t)$, respectively,

and, for the case of $\mathcal{ALC}_{L^*}(S)$,

- $(\sim C)(t)$ and $(\Delta C)(t)$ are $\sim(C(t))$ and $\Delta(C(t))$, respectively,

and, if y is a variable not occurring in $C(t)$,

- $(\forall R.C)(t)$ is $(\forall y)(R(t, y) \rightarrow C(y))$,
- $(\exists R.C)(t)$ is $(\exists y)(R(t, y) \& C(y))$.

Semantics. According to semantics for $L^*\forall(S)$, a fuzzy interpretation \mathbf{M} for this language associates

- a fuzzy set $A^{\mathbf{M}} : M \rightarrow [0, 1]$ to each atomic concept A ,
- a fuzzy relation $R^{\mathbf{M}} : M \times M \rightarrow [0, 1]$ to each atomic role R ,

and the true value for complex descriptions $\mathcal{ALC}_{L^*}(S)$ is given as follows:

$$\begin{aligned} \bar{0}^{\mathbf{M}}(a) &= 0 \\ \bar{r}^{\mathbf{M}}(a) &= r, \text{ for every } r \in S \\ (C \& D)^{\mathbf{M}}(a) &= C^{\mathbf{M}}(a) * D^{\mathbf{M}}(a) \\ (C \rightarrow D)^{\mathbf{M}}(a) &= C^{\mathbf{M}}(a) \rightarrow_* D^{\mathbf{M}}(a) \\ (\forall R.C)^{\mathbf{M}}(a) &= \inf\{R^{\mathbf{M}}(a, b) \rightarrow_* C^{\mathbf{M}}(b) : b \in M\} \\ (\exists R.C)^{\mathbf{M}}(a) &= \sup\{R^{\mathbf{M}}(a, b) * C^{\mathbf{M}}(b) : b \in M\} \end{aligned}$$

In the case of complex descriptions in $\mathcal{ALC}_{L^*}(S)$ we must add the following:

$$\begin{aligned} (\sim C)^{\mathbf{M}}(a) &= 1 - C^{\mathbf{M}}(a), \\ (\Delta C)^{\mathbf{M}}(a) &= \delta((C^{\mathbf{M}}(a))), \end{aligned}$$

where δ is the function defined on $[0, 1]$ by $\delta(x) = 1$ if $x = 1$, and $\delta(x) = 0$ otherwise.

TBox and ABox. A *fuzzy concept inclusion axiom* is a sentence of the form $\bar{r} \rightarrow (\forall x)(C(x) \rightarrow D(x))$. Following a notation similar to the used in some papers of FDLs (see for instance [21]) this sentence can be abbreviate by using the expression $\langle C \sqsubseteq D, \bar{r} \rangle$. The expression $C \equiv D$ is an abbreviation for the two axioms $\langle C \sqsubseteq D, \bar{1} \rangle$ and $\langle D \sqsubseteq C, \bar{1} \rangle$. A fuzzy *TBox* is a finite set of *fuzzy concept inclusion axioms*. A *fuzzy*

assertion axiom is a sentence of the form either $\bar{r} \rightarrow C(a)$ or $\bar{r} \rightarrow R(a, b)$. We will abbreviate these sentences with the expressions $\langle a : C, \bar{r} \rangle$ and $\langle (a, b) : R, \bar{r} \rangle$, respectively. A fuzzy *ABox* is a finite set of *fuzzy assertion axioms*. A fuzzy *KB* is a pair $\mathcal{K} = \langle \mathcal{T}, \mathcal{A} \rangle$, where the first component is a *fuzzy TBox* and the second one is a *fuzzy ABox*.

According to semantics for $L^*(\mathbf{S})$ and $L_{\sim}^*(\mathbf{S})$, given an interpretation \mathbf{M} , the interpretation of the axioms in a knowledge base \mathcal{K} is as follows:

$$\begin{aligned} \langle C \sqsubseteq D, \bar{r} \rangle^{\mathbf{M}} &= r \rightarrow_* \inf\{C^{\mathbf{M}}(x) \rightarrow_* D^{\mathbf{M}}(x) : x \in M\} \\ \langle a : C, \bar{r} \rangle^{\mathbf{M}} &= r \rightarrow_* C^{\mathbf{M}}(a^{\mathbf{M}}) \\ \langle (a, b) : R, \bar{r} \rangle^{\mathbf{M}} &= r \rightarrow_* R^{\mathbf{M}}(a^{\mathbf{M}}, b^{\mathbf{M}}) \end{aligned}$$

We say that \mathbf{M} *satisfies* a terminological or assertional axiom $\langle \Phi, \bar{r} \rangle$ iff $\langle \Phi, \bar{r} \rangle^{\mathbf{M}} = 1$. For instance, \mathbf{M} *satisfies* $\langle C \sqsubseteq D, \bar{r} \rangle$ iff $r \rightarrow_* \inf\{C^{\mathbf{M}}(x) \rightarrow_* D^{\mathbf{M}}(x) : x \in M\} = 1$, that is, pursuant to properties of standard algebras, $r \leq \inf\{C^{\mathbf{M}}(x) \rightarrow_* D^{\mathbf{M}}(x) : x \in M\}$.

4. Towards fuzzy logics for $\mathcal{ALC}_{L^*(\mathbf{S})}$ and $\mathcal{ALC}_{L_{\sim}^*(\mathbf{S})}$

$\mathcal{ALC}_{L^*(\mathbf{S})}$ and $\mathcal{ALC}_{L_{\sim}^*(\mathbf{S})}$ are defined as calculi allowing inferences in the usual sense: from a finite set of formulas Γ (a KB) a formula φ is deduced if, for any semantic interpretation evaluating all formulas of Γ as 1, φ is also evaluated as 1. It is in this framework that the algorithms defined by Straccia et al. [18,21] make sense. But the remaining question is, which is the logic that underlies this semantical calculus? In fact we want to analyze if it is possible to describe the fuzzy logic, in the sense described in Sec. 2, which is the syntactical counterpart of this semantical calculus. In this section we suggest some possible ways to answer this question. We will begin with an example to clarify our goal.

Straccia defined in [18] a FDL with a \mathcal{ALC} -like language based on the minimum t -norm (therefore, the framework is the Gödel Logic G) but taking the negation $n(x) = 1 - x$ (so the setting is G_{\sim}) and adding truth-constants in the language (now the setting is $G_{\sim}(\mathbf{S})$). Therefore, the description language is $\mathcal{ALC}_{G_{\sim}(\mathbf{S})}$ and so our first order setting is $G_{\sim}\forall(\mathbf{S})$. On the other hand, the logic $G_{\sim}\forall(\mathbf{S})$ restricted to evaluated formulas (i.e., formulas of the form $\varphi = \bar{r} \rightarrow \psi$, where ψ does not contain truth constants) satisfies the following completeness theorem:

$$\Gamma \vdash_{G_{\sim}\forall(\mathbf{S})} \varphi \text{ iff } \Gamma \models_{[0,1]_{G_{\sim}(\mathbf{S})}} \varphi, \quad (1)$$

where $\Gamma \cup \{\varphi\}$ is a set of evaluated formulas of the logic $G_{\sim}\forall(\mathbf{S})$, and $[0,1]_{G_{\sim}(\mathbf{S})}$ is the standard algebra $\langle [0,1], \min, \max, \rightarrow_G, \sim, \{\bar{r} \mid r \in S\}, 0, 1 \rangle$. The proof is a simplified version of [7, Theorem 11]. Notice that as a consequence of that theorem and from the fact that the semantic for $\mathcal{ALC}_{G_{\sim}(\mathbf{S})}$ coincides with the semantic over the chain $[0,1]_{G_{\sim}(\mathbf{S})}$, the logic $\mathcal{ALC}_{G_{\sim}(\mathbf{S})}$ coincides with the fragment of $G_{\sim}\forall(\mathbf{S})$ restricted to the language of $\mathcal{ALC}_{G_{\sim}(\mathbf{S})}$. In other words, $\Gamma \vdash_{\mathcal{ALC}_{G_{\sim}(\mathbf{S})}} \varphi$ iff $\Gamma \models_{[0,1]_{G_{\sim}(\mathbf{S})}} \varphi$, that it is an interesting theoretical result since it gives an equivalence between $\mathcal{ALC}_{G_{\sim}(\mathbf{S})}$ and the semantics consequence relation on $[0,1]_{G_{\sim}(\mathbf{S})}$. So, we can state that $\mathcal{ALC}_{G_{\sim}(\mathbf{S})}$ is the fuzzy logic which is the counterpart of the semantic system used by Straccia in [18].

The Gödel t -norm is an example of strict t -norm since it satisfies $x * y = 0$ implies $x = 0$ or $y = 0$. This property seems to be essential for proving the previous completeness theorem. As future work we plan to analyze whether or not the theorem can be gen-

eralized for logics $\mathcal{ALC}_{L_{\sim}}(\mathbf{S})$ where $*$ is some other strict t-norm. However, a completeness theorem such as (1) does not hold for all fuzzy logics $L^*\forall(\mathbf{S})$ and $L_{\sim}^*\forall(\mathbf{S})$, therefore the method we used to prove the equivalence between $\mathcal{ALC}_{G_{\sim}}(\mathbf{S})$ and the semantics consequence relation on $[0, 1]_{G_{\sim}}(\mathbf{S})$ cannot be used.² In such situations we need to study the fragment defining the logic $\mathcal{ALC}_L(\mathbf{S})$ directly trying to prove that this fragment is axiomatizable, decidable and computational tractable.

5. Concluding Remarks

This paper is a first step on the direction proposed by Hájek concerning to analyze the relationships between FDLs and the recent developments in mathematical fuzzy logics. The main contributions of our approach is the use of truth constants in the language of description. This choice is oriented to search for the syntactical counterpart of the semantic calculi used in most of works dealing with FDLs.

In classical DLs, the logical system is always Classical Logic and each DL depends only on the language considered. Thus the main theoretical problems are the relationships between expressiveness and either decidability or computational complexity of the considered language. In our approach to FDLs the situation is more complex since we need to choose both a t-norm and a countable subalgebra of the corresponding standard algebra for each language. Therefore the theoretical problems mentioned above appear for each one of the languages since they depends on the logical system underlying the FDL at hand. In this sense, these problems could take benefit from the recent advances on Mathematical Fuzzy Logic for solving them. The idea is that because we work with fragments of first order logics some of general results coming from mathematical fuzzy logics could be applied to particular fragments. In some cases a direct study of the corresponding fragment seems to be necessary. Moreover, as in the classical DLs, it is also possible to translate the FDLs into a modal setting but this need previous studies of modal logics over different multiple-valued (fuzzy) logics. This is a work that has taken only small attention in the past and need further development to be used in the framework of Fuzzy Description Logics. Finally we want to remark that many applications uses a finite number of truth values. Thus to study the cases where the truth value set is finite could be quite interesting.

Acknowledgments

This work has been partially supported by the Spanish project MÚLOG TIN2007-68005-C04-01/04, the CSIC grant 200750I005 and the Generalitat de Catalunya grant 2005-SGR-00093. The authors gratefully acknowledge the many helpful suggestions of Eva Armengol to improve this work.

²An example of the non-applicability is the $\mathcal{ALC}_L(\mathbf{S})$ where it is well known that the first order tautologies of the canonical Łukasiewicz chain are not recursively axiomatizable.

References

- [1] F. Baader, I. Horrocks, and U. Sattler. Description logics. In F. van Harmelen, V. Lifschitz, and B. Porter, editors, *Handbook of Knowledge Representation*, pages 135–179. Elsevier, 2007.
- [2] R. Cignoli, F. Esteva, L. Godo, and A. Torrens. Basic fuzzy logic is the logic of continuous t-norms and their residua. *Soft Computing*, 4(2):106–112, 2000.
- [3] F. Esteva, J. Gispert, L. Godo, and C. Noguera. Adding truth-constants to logics of continuous t-norms: Axiomatization and completeness results. *Fuzzy Sets and Systems*, 158(6):597–618, March 2007.
- [4] F. Esteva and L. Godo. Monoidal t-norm based logic: towards a logic for left-continuous t-norms. *Fuzzy Sets and Systems*, 124:271–288, 2001.
- [5] F. Esteva, L. Godo, P. Hájek, and M. Navara. Residuated fuzzy logics with an involutive negation. *Archive for Mathematical Logic*, 39(2):103–124, 2000.
- [6] F. Esteva, L. Godo, and F. Montagna. Axiomatization of any residuated fuzzy logic defined by a continuous t-norm. In *Fuzzy Sets and Systems - IFSA 2003, 10th International Fuzzy Systems Association World Congress, Istanbul, Turkey, June 30 - July 2, 2003, Proceedings*, pages 172–179, 2003.
- [7] F. Esteva, L. Godo, and C. Noguera. On completeness results for predicate łukasiewicz, product, Gödel and nilpotent minimum logics expanded with truth-constants. *Mathware & Soft Computing*, 14(3):223–246, 2007.
- [8] T. Flaminio and E. Marchioni. T-norm-based logics with an independent involutive negation. *Fuzzy Sets and Systems*, 157:3125–3144, 2006.
- [9] P. Hájek. *Metamathematics of fuzzy logic*, volume 4 of *Trends in Logic—Studia Logica Library*. Kluwer Academic Publishers, Dordrecht, 1998.
- [10] P. Hájek. Making fuzzy description logic more general. *Fuzzy Sets and Systems*, 154(1):1–15, 2005.
- [11] P. Hájek. What does mathematical fuzzy logic offer to description logic? In Elie Sanchez, editor, *Fuzzy Logic and the Semantic Web, Capturing Intelligence*, chapter 5, pages 91–100. Elsevier, 2006.
- [12] S. Jenei and F. Montagna. A proof of standard completeness for Esteva and Godo’s logic MTL. *Studia Logica*, 70:183–192, 2002.
- [13] J. Pavelka. On fuzzy logic. I. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 25(1):45–52, 1979. Many-valued rules of inference.
- [14] J. Pavelka. On fuzzy logic. II. Enriched residuated lattices and semantics of propositional calculi. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 25(2):119–134, 1979.
- [15] J. Pavelka. On fuzzy logic. III. Semantical completeness of some many-valued propositional calculi. *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, 25(5):447–464, 1979.
- [16] P. Savický, R. Cignoli, F. Esteva, L. Godo, and C. Noguera. On product logic with truth-constants. *Journal of Logic and Computation*, 16(2):205–225, 2006.
- [17] G. Stoilos, G. Stamou, J.Z. Pan, V. Tzouvaras, and I. Horrocks. Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research*, 30(8):273–320, 2007.
- [18] U. Straccia. Reasoning within fuzzy description logics. *J. Artif. Intell. Res.*, 14:137–166, 2001.
- [19] U. Straccia. Fuzzy alc with fuzzy concrete domains. In *Proceedings of the International Workshop on Description Logics (DL-05)*, pages 96–103, Edinburgh, Scotland, 2005. CEUR.
- [20] U. Straccia. A fuzzy description logic for the semantic web. In Elie Sanchez, editor, *Fuzzy Logic and the Semantic Web, Capturing Intelligence*, chapter 4, pages 73–90. Elsevier, 2006.
- [21] U. Straccia and F. Bobillo. Mixed integer programming, general concept inclusions and fuzzy description logics. *Mathware & Soft Computing*, 14(3):247–259, 2007.
- [22] C. B. Tresp and R. Molitor. A description logic for vague knowledge. Technical Report RWTH-LTCS Report 98-01, Aachen University of Technology, 1998.
- [23] J. Yen. Generalizing term subsumption languages to fuzzy logic. In *Proc. of the 12th IJCAI*, pages 472–477, Sidney, Australia, 1991.

An interval-based approach for fault isolation and identification in continuous dynamic systems

Esteban R. GELSO, Sandra M. CASTILLO , and Joaquim ARMENGOL
IliA - Universitat de Girona, Campus de Montilivi, Girona, E-17071, Spain
{*esteban.gelso, sandra.castillo, joaquim.armengol*}@udg.edu

Abstract. Diagnosis of faults is a very important task because a fault can lead to a reduction in performance, or even to break-downs or catastrophes. In this paper, a diagnosis system, which takes into account the uncertainties in the model and measurements by means of an interval scheme, is shown. It states the fault detection, fault isolation and fault identification problems as a constraint satisfaction problem (CSP) with continuous domain. The fault isolation scheme which consists of hypothesis generation and refinement uses the fault detection results, and the quantitative fault identification scheme refines the fault hypothesis test and estimates the fault magnitude. As a difference to more traditional approaches, in this paper we use overconstrained subsystems of the model, ARRs, to tackle the fault identification problem in smaller subproblems, and for the fault identification, instead of the least squares estimation, the interval-based consistency techniques prune the initial domains of the parameters associated with the fault hypothesis. In order to illustrate the different steps of the proposed approach, an application example composed by a well known hydraulic plant is presented.

Keywords. Model-based, Fault isolation, Fault identification, uncertainty, consistency techniques

Introduction

Diagnosis of faults is a very important task because a fault can lead to a reduction in performance, or even to break-downs or catastrophes. One approach to diagnosis is Model Based Diagnosis (MBD), which is based on comparing observations of the actual behavior of the process and the behavior of a model of the process. Two research communities have used the MBD approach in parallel. (i) the FDI (Fault Detection and Isolation) community, formed by researchers with a background in control systems engineering, has used the analytical redundancy approach [1,2], and (ii), the Diagnosis community (DX), from the fields of computer science and artificial intelligence, has made a contribution, among other approaches, using the consistency-based logical approach [3,4].

In this paper, a diagnosis system, which takes into account the uncertainties in the model and measurements by means of an interval scheme, will be shown. It states the fault detection, isolation, and identification problem as a constraint satisfaction problem (CSP) with continuous domain. CSPs with continuous domains are very common in the

real world, and a well known category of them is that of linear programming problems, where constraints must be linear inequalities forming a convex region [5]. In this paper, the resolution of the CSP is performed by combining interval methods and constraint satisfaction techniques (See Section 1).

The diagnosis approach is presented in Section 2, and consists of three main blocks: (i) a fault detector which monitors the difference in the observed and expected behavior (residual), (ii) a fault isolation scheme which consists of hypothesis generation and refinement using the fault detection results of the analytical redundancy relations (ARR) of the system, and (iii) the quantitative fault isolation and identification scheme to refine the fault hypothesis test and to estimate the fault magnitude. This approach is stated as in [6,7], but in this paper we use overconstrained subsystems of the model, ARRs, to tackle the fault identification problem in smaller subproblems. Moreover, for the fault identification, instead of using the least squares estimation, interval-based consistency techniques prune the initial domains of the parameters associated with the fault hypothesis.

An application example is presented in Section 3, and finally, some conclusions are given at the end of the paper.

1. Interval-Based Consistency Techniques

One way to represent the uncertainties in the model and measurements is by means of an interval scheme, where the parameters and measurements take on values within defined intervals. In some situations, we can say about the values of a variable, x is bounded by values $[\underline{x}, \bar{x}]$, $\underline{x} \leq \bar{x}$ with certainty but the distribution within this interval may be unknown [8]. This situation occurs often in measurement practice when, for a sensor, the only information we may have are the bounds on its error values.

Many engineering problems (e.g., parameter and state estimation, robust control design problems) can be formulated in a logical form by means of some kind of first order predicate formulas: formulas with the logical quantifiers (universal and existential), a set of real continuous functions (equalities and inequalities), and variables ranging over real interval domains.

As defined in [9], a numerical constraint satisfaction problem is a triple $\mathcal{CSP} = (\mathcal{V}, \mathcal{D}, \mathcal{C}(x))$ defined by

1. a set of numeric variables $\mathcal{V} = \{x_1, \dots, x_n\}$,
2. a set of domains $\mathcal{D} = \{D_1, \dots, D_n\}$ where D_i , a set of numeric values, is the domain associated with the variable x_i ,
3. a set of constraints $\mathcal{C}(x) = \{C_1(x), \dots, C_m(x)\}$ where a constraint $C_i(x)$ is determined by a numeric relation (equation, inequality, inclusion, etc.) linking a set of variables under consideration.

Interval-based consistency techniques can be used to contract the domains of the involved variables by removing inconsistent values [10,11,12]. In particular for the fault detection application, they are used to guarantee that there is a fault when there is no solution that can be found for the CSP problem, i.e. the observed behavior and the model are inconsistent. Moreover, for the fault isolation/identification application, they are used to reject a hypothesis test of a set of behavioral modes. In this manner, it helps to refine the fault hypothesis set and to estimate the fault magnitude.

The algorithms that are based on consistency techniques are actually "branch and prune" algorithms, i.e., algorithms that can be defined as an iteration of two steps [10]:

1. Pruning the search space by reducing the intervals associated with the variables until a given consistency property is satisfied.
2. Generating subproblems by splitting the domains of a variable

These techniques can be applied to nonlinear dynamic systems, and their results are not sensitive to strong nonlinearities or nondifferentiabilities in the dynamic system [13].

Most interval constraint solvers are based on either hull-consistency (also called 2B-consistency) or box-consistency, or a variation of them [11]. Box-consistency tackles the problem of hull-consistency for variables with many occurrences in a constraint. The aforementioned techniques are said to be local: each reduction is applied over one domain with respect to one constraint. Better pruning of the variable domains may be achieved if, complementary to a local property, some global properties are also enforced on the overall constraint set.

In this paper, the solution of the fault detection/identification *CSP* is performed by using the solver RealPaver [14], and BC4 consistency.

2. Diagnosis System

The diagnosis approach is illustrated in Fig. 1. The first step is the robust fault detection, which tracks the nominal (non-faulty) system dynamics using the interval-based consistency technique, and detects a fault when no solution is found in the CSP [15].

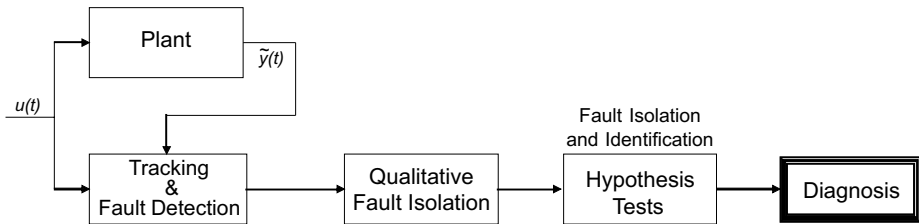


Figure 1. Diagnosis scheme.

As a first step of the fault detection, analytical redundancy relations (ARR) are deduced off-line by means of Structural Analysis, from overconstrained subsystems of the model. Once they are designed, the fault detection procedure checks on-line the consistency of the observation with respect to every ARR.

When discrepancies occur between the modeled behavior and the observations, the fault isolation procedure identifies the diagnosis candidates through the fault signature matrix. This matrix expresses the theoretical influence of the faults onto the residuals [16]. To reduce adequately the set of diagnoses and to minimize the time required for fault identification, qualitative information (obtained from the fault models and their effect on the system) can be integrated into the fault signature matrix [17,18].

2.1. Quantitative Fault Isolation And Identification

Quantitative fault isolation and identification is the final step in the fault diagnosis procedure. The objective of this task is: (i) to refine the fault hypothesis set, because the qualitative fault isolation can not return an unique fault candidate, and (ii), to estimate the magnitude of the fault.

The faults may be represented as unknown extra inputs acting on the system (additive faults), or as changes in some plant parameters (multiplicative faults) [2]. The constraints in the model can be rewritten by including knowledge about the behavioral modes. Additive faults can represent plant leaks or biases in sensors, for example, and multiplicative faults can represent clogging or deterioration of plant equipment, for example. In a case of a sensor bias, the equation describing the relation between the measured value \tilde{y} and the physical variable y can be expressed as $\tilde{y} = y + f$, where f is the constant bias of the sensor. Similarly, for a linear fault in the sensor, the equation can be expressed as $\tilde{y} = a y + b$, where a and b are the constant parameters of this fault.

The fault isolation and identification methodology uses the framework of structured hypothesis tests proposed in [6]. The hypothesis tests used are binary, i.e., the task is to test one null hypothesis against one alternative hypothesis. The null hypothesis and the alternative hypothesis can be written as

$$\begin{aligned} H_k^0: & F_p \in M_k \\ & \text{"some behavioral mode in } M_k \text{ can explain measured data"} \\ H_k^1: & F_p \in M_k^C \\ & \text{"no behavioral mode in } M_k \text{ can explain measured data"} \end{aligned}$$

Where F_p denote the present behavioral mode, and M_k is the set of behavioral modes for the k th hypothesis test. When H_k^0 is rejected, H_k^1 is assumed to be true, hence, the present behavioral mode can not belong to M_k . Further, when H_k^0 is not rejected, in most cases nothing is assumed. Each hypothesis test contributes with a piece of information regarding which behavioral modes that can be matched to the data or not.

Intervals of the fault parameters that are consistent with the measurements of the system are estimated using the interval-based consistency techniques applied to the ARRs in which the fault was detected. The initial domains for all the parameters are substituted by the nominal intervals, except for the parameter or parameters associated with the fault hypothesis. If no solution is found in the CSP of the k th fault hypothesis, then H_k^0 is rejected and we may drop that corresponding hypothesis.

3. An Application To Coupled Water Tanks

A non-linear dynamical example of a system based on two coupled water tanks will be used to explain, mainly, the quantitative analysis for fault isolation and identification. Figure 2 shows a schematic drawing of the system.

The system is composed of two tanks, $T1$ and $T2$, a valve, $V1$, and a controller, $PI1$, which receives the current level of $T2$ as the input, and controls a valve, $V1$, which regulates the flow of water to $T1$.

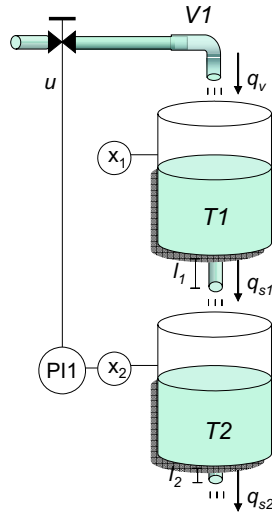


Figure 2. Diagram of the coupled water tanks system.

3.1. Model Equations

The system is described by the elementary analytical relations shown in Table 1.

Table 1. Elementary analytical relations of the two coupled tanks system

| | Elementary Relations | Component |
|-----|---|------------------------|
| (a) | $q_v = k u^3$ | Valve |
| (b) | $S_1 \frac{dx_1}{dt} = q_v - q_{s1}$ | Upper tank |
| (c) | $q_{s1} = k_{s1} \sqrt{x_1 + l_1}$ | Output pipe upper tank |
| (d) | $S_2 \frac{dx_2}{dt} = q_{s1} - q_{s2}$ | Lower tank |
| (e) | $q_{s2} = k_{s2} \sqrt{x_2 + l_2}$ | Output pipe lower tank |
| (f) | $\tilde{u} = u$ | D/A converter |
| (g) | $\tilde{x}_1 = x_1$ | x_1 sensor |
| (h) | $\tilde{x}_2 = x_2$ | x_2 sensor |

The terms q_v , q_{s1} and q_{s2} denote the volumetric flows, x_1 and x_2 are the heights of the water in tanks $T1$ and $T2$, respectively, and u is the output signal of the controller. The variables u , q_v , q_{s1} , q_{s2} , x_1 , and x_2 are unknown, \tilde{u} , \tilde{x}_1 , and \tilde{x}_2 are known variables obtained from sensors, and k , k_{s1} , k_{s2} , S_1 , S_2 , l_1 and l_2 are the constant parameters of the system.

All the variables and parameters are considered as intervals for the consistency test using the solver RealPaver. The intervals for the measurements include the accuracy error and the noise level, and for the parameters, they were estimated using a strong consistency technique with data in a fault free scenario.

The faults considered, are sensor faults, actuator faults and process faults (leakages and clogging in the output pipes of $T1$ and $T2$). In this example we assume that only single, abrupt faults occur. In the case of incipient faults, for the fault identification, the

methodology explained in this article can also be applied. The model of an incipient fault can be represented as in [19], adding a drift term to the nominal parameter value.

The following behavioral modes are considered:

- Gain-fault in the height sensor of $T1$. The model corresponding to this behavioral mode is obtained by using Table 1, but replacing equation (g) with $\tilde{x}_1 = x_1(1 - f_1)$, where f_1 represents a gain fault and is $0 < f_1 < 1$ when this fault is present.
- Gain-fault in the height sensor of $T2$. This behavioral mode is obtained by using Table 1, but replacing equation (h) with $\tilde{x}_2 = x_2(1 - f_2)$, where f_2 represents a gain fault and is $0 < f_2 < 1$ when this fault is present.
- Bias in the D/A converter. This behavioral mode is obtained from the fault free model, but replacing equation (f) with $\tilde{u} = u + f_3$, where f_3 is a constant bias and $f_3 \neq 0$ when this fault is present.
- Leakage in $T1$. As explained before, but replacing equation (b) with $S_1 \frac{dx_1}{dt} = q_v - q_{s1} - f_4 \sqrt{x_1}$. When this fault is present, $f_4 \neq 0$ and $f_4 > 0$.
- Clogging fault in the output pipe of $T1$. This behavioral mode is obtained from the fault free model, but replacing equation (c) with $q_{s1} = k_{s1}(1 - f_5) \sqrt{x_1 + l_1}$. When this fault is present, $f_5 \neq 0$ and $0 < f_5 < 1$.
- Leakage in $T2$. This behavioral mode is obtained from the fault free model, but replacing equation (d) with $S_2 \frac{dx_2}{dt} = q_{s1} - q_{s2} - f_6 \sqrt{x_2}$. When this fault is present, $f_6 \neq 0$ and $f_6 > 0$.
- Clogging fault in the output pipe of $T2$. This behavioral mode is obtained from the fault free model, but replacing equation (e) with $q_{s2} = k_{s2}(1 - f_7) \sqrt{x_2 + l_2}$. When this fault is present, $f_7 \neq 0$ and $0 < f_7 < 1$.

Two different ARR_s were obtained with the structural analysis, which are minimal with respect to the set of constraints used in the model. They corresponds to the mass balance of each tank. Equations $\{a, b, c, f, g\}$ are involved in ARR₁, and equations $\{c, d, e, g, h\}$ are involved in ARR₂. Considering the behavioral modes, the fault signature matrix can be deduced as shown in Table 2.

Table 2. Fault signature matrix.

| | f_1 | f_2 | f_3 | f_4 | f_5 | f_6 | f_7 |
|------|-------|-------|-------|-------|-------|-------|-------|
| $r1$ | 1 | 0 | 1 | 1 | 1 | 0 | 0 |
| $r2$ | 1 | 1 | 0 | 0 | 1 | 1 | 1 |

3.2. Simulation Results

To illustrate the performance of the diagnosis system, a faulty scenario involving a clogging fault in the output pipe of $T1$, f_5 , is considered. The values of the measured variables are collected once per 20 seconds, and the sensors of x_1 and x_2 have an uncertainty of ± 0.5 cm. The parameters of the model are also represented by intervals to take into account any associated uncertainty in the model. Fig. 3 shows the measured signals for this scenario. The fault has a magnitude of 10% of k_{s1} , and begins at sample 225.

A sliding time window of length 20 samples is used for the fault detection. The fault is detected at sample 232 using ARR₁, and the fault isolation, based on the fault signature (Table 2), shows the initial list of four candidates, $\{f_1, f_3, f_4, f_5\}$. For the ARR₂ case,

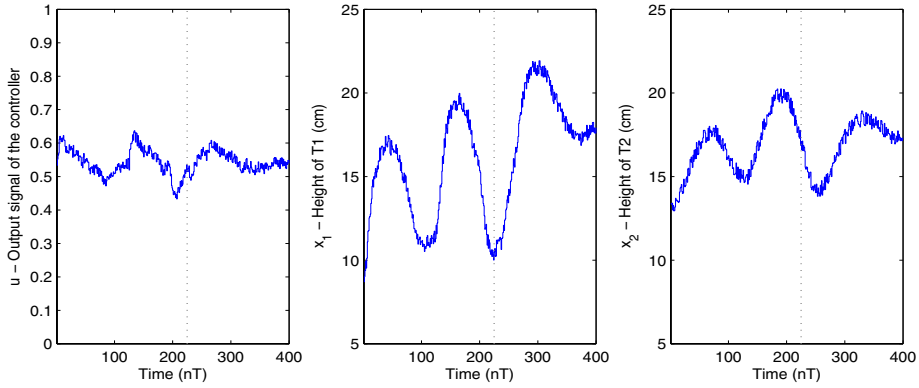


Figure 3. Measured signals when a clogging fault in the output pipe of $T1$ is present.

the fault is detected at sample 242, and then, two of the initial candidates are dropped, $\{f_3, f_4\}$.

To show in this scenario the efficiency of the quantitative fault isolation and identification, we didn't take into account the signs in the fault signature matrix, which would lead to a faster diagnosis.

The quantitative fault isolation and identification starts for each ARR whose fault detection alarm was activated, for ARR1 it starts at sample 232 and for ARR2 it starts at sample 242. The estimation of parameters f_1 and f_5 using ARR1 and ARR2 is represented in Figs. 4 and 5. The final estimated interval of parameter f_1 is given by the intersection of the two intervals obtained by using ARR1 (Fig. 4a) and ARR2 (Fig. 4b), and is depicted in Fig. 4c. Similarly, the final estimated interval of parameter f_5 is given by the intersection of the two intervals obtained by using ARR1 (Fig. 5a) and ARR2 (Fig. 5b), and is depicted in Fig. 5c.

Where the intersection of two intervals $X = [\underline{x}, \bar{x}]$ and $Y = [\underline{y}, \bar{y}]$ is defined by

$$X \cap Y \triangleq \{z \in \mathbb{R} \mid z \in X \text{ and } z \in Y\},$$

and satisfies

$$\begin{aligned} X \cap Y &= [\max\{\underline{x}, \underline{y}\}, \min\{\bar{x}, \bar{y}\}] \quad \text{if } \max\{\underline{x}, \underline{y}\} \leq \min\{\bar{x}, \bar{y}\}, \\ &= \emptyset \quad \text{otherwise.} \end{aligned}$$

In the example, at sample 270, the fault hypothesis of a gain-fault in the height sensor of $T1$ (f_1) is rejected after no solution was found in the CSP of the ARR2. At sample 332, the final estimated interval of parameter f_5 is equal to $[0.0525, 0.1598]$, and as it was expected, it includes the real value of the fault, 0.1.

Therefore the clogging fault in the output pipe of $T1$ is the only behavioral mode that can explain the behavioral of the system.

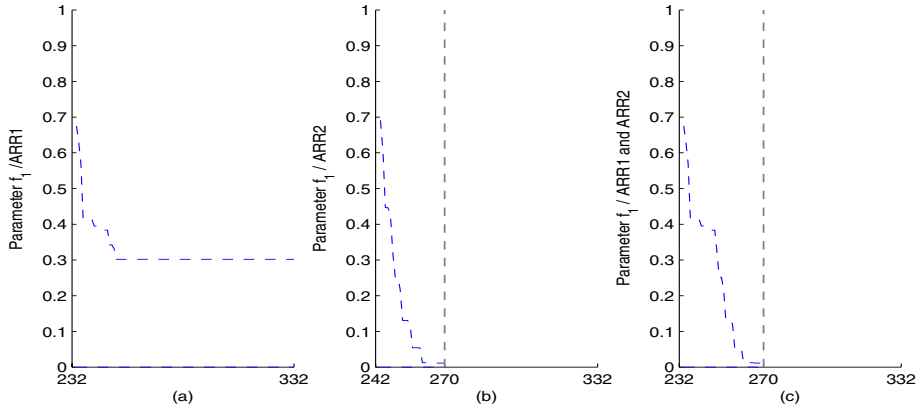


Figure 4. Fault identification of parameter f_1 .

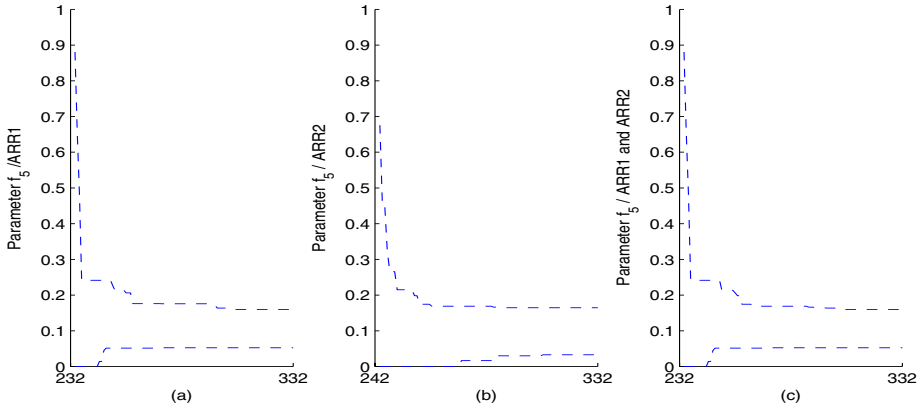


Figure 5. Fault identification of parameter f_5 .

4. Conclusion

As many engineering problems, fault diagnosis problems can be stated as a CSP in continuous domains. The work presented in this paper is an example of that. The diagnosis approach presented follows three main blocks: (i) a fault detector which monitors the difference in the observed and expected behavior (residual), (ii) a fault isolation scheme which consists of hypothesis generation and refinement using the fault detection results of the analytical redundancy relations (ARR) of the system, and (iii) the quantitative fault isolation and identification scheme to refine the fault hypothesis test and to estimate the fault magnitude.

The use of overconstrained subsystems of the model, ARRs, leads the approach to tackle the fault identification problem in smaller subproblems which can reduce the complexity of the resolution problem. CSPs are used also to represent the fault identification problem, and hence, to calculate the fault magnitude.

Future work will consist of analyzing multiple fault case, and the complexity of the fault identification technique with respect to the parameter vector dimension. The influence of the sliding window length must be studied in future also.

5. Acknowledgements

This work has been funded by the Spanish Government (Plan Nacional de Investigación Científica, Desarrollo e Innovación Tecnológica, Ministerio de Educación y Ciencia) through the coordinated research project grant No. DPI2006-15476-C02-02, by the grant No. 2005SGR00296 and the Departament d'Innovació, Universitats, i Empresa of the Government of Catalonia.

References

- [1] R. J. Patton, P. M. Frank, and R. N. Clark, *Issues of fault diagnosis for dynamic systems*. Springer, 2000.
- [2] J. J. Gertler, *Fault Detection and Diagnosis in Engineering Systems*. Marcel Dekker, 1998.
- [3] R. Reiter, "A theory of diagnosis from first principles," *Artificial Intelligence*, vol. 32, no. 1, pp. 57–95, 1987.
- [4] W. Hamscher, L. Console, and J. de Kleer, *Readings in Model-Based Diagnosis*. San Francisco, CA, USA: Morgan Kaufman, 1992.
- [5] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*. Prentice-Hall, 2003.
- [6] M. Nyberg, "Model-based diagnosis of an automotive engine using several types of fault models," *IEEE Transaction on Control Systems Technology*, vol. 10, pp. 679–689, September 2002.
- [7] E. Manders, S. Narasimhan, G. Biswas, and P. Mosterman, "A combined qualitative/quantitative approach for fault isolation in continuous dynamic systems," in *4th Symposium on Fault Detection, Supervision and Safety for Technical Processes*, (Budapest, Hungary), pp. 1074–1079, 2000.
- [8] S. Ferson, L. Ginzburg, V. Kreinovich, and J. Lopez, "Absolute bounds on the mean of sum, product, etc.: A probabilistic extension of interval arithmetic," *Applied Mathematical Sciences*, vol. 1, no. 9, pp. 395–440, 2007.
- [9] S. Shary, "A new technique in systems analysis under interval uncertainty and ambiguity," *Reliable Computing*, vol. 8, pp. 321–418, 2002.
- [10] H. Collavizza, F. Delobel, and M. Rueher, "Comparing partial consistencies," *Reliable Computing*, vol. 5, pp. 213–228, 1999.
- [11] F. Benhamou, F. Goualard, L. Granvilliers, and J. F. Puget, "Revising hull and box consistency," in *Proceedings of the International Conference on Logic Programming*, (Las Cruces, NM), pp. 230–244, 1999.
- [12] J. Cruz and P. Barahona, "Maintaining global hull consistency with local search for continuous CSPs," in *COCOS*, pp. 178–193, 2002.
- [13] L. Jaulin, "Consistency techniques for the localization of a satellite," in *COCOS*, pp. 157–170, 2002.
- [14] L. Granvilliers and F. Benhamou, "Algorithm 852: Realpaver: an interval solver using constraint satisfaction techniques," *ACM Trans. Math. Softw.*, vol. 32, no. 1, pp. 138–156, 2006.
- [15] E. Gelso, S. Castillo, and J. Armengol, "Robust fault detection using consistency techniques for uncertainty handling," in *IEEE International Symposium on Intelligent Signal Processing, WISP 2007*, (Alcalá de Henares, Spain), 2007.
- [16] M. O. Cordier, P. Dague, F. Lévy, J. Montmain, M. Staroswiecki, and L. Travé-Massuyès, "Conflicts versus analytical redundancy relations: a comparative analysis of the model based diagnosis approach from the artificial intelligence and automatic control perspectives," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 34, pp. 2163–2177, October 2004.
- [17] G. Calderón-Espinoza, J. Armengol, J. Vehí, and E. Gelso, "Dynamic diagnosis based on interval analytical redundancy relations and signs of the symptoms," *AI Communications*, vol. 20, no. 1, pp. 39–47, 2007.
- [18] P. J. Mosterman and G. Biswas, "Diagnosis of continuous valued systems in transient operating regions," *IEEE Trans. on Systems, Man and Cybernetics, Part A*, vol. 29, pp. 554–565, November 1999.
- [19] I. Roychoudhury, G. Biswas, and X. Koutsoukos, "A bayesian approach to efficient diagnosis of incipient faults," in *17th International Workshop on Principles of Diagnosis DX*, pp. 243–264, 2006.

Effects of orness and dispersion on WOWA sensitivity

Vicenç TORRA ^{a,1}

^a *IIIA-CSIC, Institut d'Investigació en Intel·ligència Artificial*

Abstract. The WOWA operator is an aggregation operator that generalizes the weighted mean and the OWA operator. In this paper we study the sensitivity of this operator with respect to the interpolation method used in its calculation. We analyze the sensitivity with respect to the orness and dispersion (entropy) of the weighting vector.

Keywords. Aggregation operators, WOWA, orness, dispersion

1. Introduction

Information fusion and aggregation operators have received special attention in the last years. [8] defines aggregation operators as the particular mathematical functions that combine N values in a given domain D and return a value in the same domain. Such operators are required to satisfy unanimity, and, when D is an ordinal scale, monotonicity.

In the case of numerical data (e.g., when D is a subset of the real line), a few operators have been defined. The arithmetic mean and the weighted mean are the most well known ones.

In 1988, Yager [9] defined the OWA (Ordered Weighted Averaging) operator. This operator corresponds to a linear combination of order statistics. Later, Torra [4,5] defined the WOWA operator (the Weighted OWA operator) as a generalization of the weighted mean and the OWA operator.

A main characteristic of the WOWA operator is that it is defined in terms of two weighting vectors (recall that both the weighted mean and the OWA use a single one). While from a formal point of view, the two vectors are equivalent (they are defined in terms of positive weights that add to one), they have a different interpretation. One of the weighting vectors corresponds to the one in the weighted mean and, thus, they represent the importance or reliability of the sources. The other vector corresponds to the one of the OWA and, thus, it permits the user to assign importance to the values. That is, the user can give more importance to larger values than to the lower ones, or, in the contrary, the user can give more importance to lower values than to larger ones. Having the two weighting vectors, the user can represent both types of background information about the data being aggregated and the data sources that supply these data. For a description

¹Corresponding Author: IIIA-CSIC; Campus UAB s/n; E-08193 Bellaterra, Catalonia; E-mail: vtorra@iiia.csic.es

of some situations that can be modeled using the WOWA operator (and that cannot be represented using the OWA or the weighted mean see e.g. [8]).

The definition of the WOWA operator relies on a fuzzy quantifier that is built from the weighting vector used in the OWA. Nevertheless, given a weighting vector, the definition does not settle uniquely the fuzzy quantifier (except for some particular weighting vectors). In particular, the fuzzy quantifier depends on an interpolation method. Accordingly, different interpolation methods lead to different fuzzy quantifiers.

In a recent work [7], we have considered the problem of WOWA sensitivity with respect to different interpolation methods. In this paper we review some of the results obtained and we further study the sensitivity of the WOWA with respect to the weights. In particular, we study the influence of their dispersion (computed in terms of its entropy, as usual [9,8]) on the OWA weights.

The structure of the paper is as follows. In Section 2 we review the definitions of the WOWA operator as well as a few results that are needed in the rest of the paper. Then, in Section 3 we study the effect of the entropy on the sensitivity of the WOWA. The paper finishes with the conclusions and outlines some future lines for research.

2. Preliminaries

We review in this section a few definitions that are used later on in this paper. In particular, we review the definition of the WOWA operator and the definitions of orness and dispersion for weighting vectors.

As stated above, the WOWA operator was defined to generalize the weighted mean and the OWA. This is achieved by means of two weighting vectors. We will denote them by p and w . One corresponds to the weighting vector used in the weighted mean (we use p in this paper), while the other corresponds to the one used in the OWA operator (we use w).

Definition 1 [5] *Let p and w be two weighting vectors of dimension N ; then, a mapping $WOWA: \mathbb{R}^N \rightarrow \mathbb{R}$ is a Weighted Ordered Weighted Averaging (WOWA) operator of dimension N if*

$$WOWA_{p,w}(a_1, \dots, a_N) = \sum_{i=1}^N \omega_i a_{\sigma(i)},$$

where σ is defined as a permutation of $\{1, \dots, N\}$ such that $a_{\sigma(i)}$ is the i th largest element in the collection a_1, \dots, a_N , and the weight ω_i is defined as

$$\omega_i = w^*\left(\sum_{j=i}^N p_{\sigma(j)}\right) - w^*\left(\sum_{j<i} p_{\sigma(j)}\right),$$

with w^* being a nondecreasing function that interpolates the points

$$\{(i/N, \sum_{j=i}^N w_j)\}_{i=1,\dots,N} \cup \{(0, 0)\}.$$

The function w^* is required to be a straight line when the points can be interpolated in this way.

As stated above, the WOWA operator generalizes the weighted mean and the OWA operator. In particular, when $w = (1/N, \dots, 1/N)$ we have $WOWA_{\mathbf{p}, \mathbf{w}}(a_1, \dots, a_N) = WM_{\mathbf{p}}(a_1, \dots, a_N)$ and when $p = (1/N, \dots, 1/N)$ we have $WOWA_{\mathbf{p}, \mathbf{w}}(a_1, \dots, a_N) = OWA_{\mathbf{w}}(a_1, \dots, a_N)$. In the particular case that $p = w = (1/N, \dots, 1/N)$ we have that the WOWA reduces to an arithmetic mean.

There is an equivalent definition in terms of fuzzy quantifiers. In fact, the function w^* can be understood as a fuzzy quantifier, and then we can define the WOWA from the function w^* and the weighting vector p . This definition of the WOWA is equivalent to the OWA with importances defined by Yager [10]. We review this definition below.

Definition 2 A function $Q: [0,1] \rightarrow [0,1]$ is a regular nondecreasing fuzzy quantifier if (i) $Q(0) = 0$; (ii) $Q(1) = 1$; and (iii) $x > y$ implies $Q(x) \geq Q(y)$.

Definition 3 Let Q be a regular nondecreasing fuzzy quantifier, and let \mathbf{p} be a weighting vector of dimension N ; then, a mapping $WOWA: \mathbb{R}^N \rightarrow \mathbb{R}$ is a Weighted Ordered Weighted Averaging (WOWA) operator of dimension N if

$$WOWA_{\mathbf{p}, Q}(a_1, \dots, a_N) = \sum_{i=1}^N \omega_i a_{\sigma(i)},$$

where σ is defined as in the case of the OWA, and the weight ω_i is defined as

$$\omega_i = Q\left(\sum_{j \leq i} p_{\sigma(j)}\right) - Q\left(\sum_{j < i} p_{\sigma(j)}\right).$$

In [7] we introduced another definition of the WOWA operator that permits to establish an explicit connection between the two definitions above.

Definition 4 [7] Let w be a weighting vector of dimension N with weights $w = (w_1 \dots w_N)$, let I denote a particular interpolation method, and let $I(w)$ be the function that interpolates the points $\{(i/N, \sum_{j \leq i} w_j)\}_{i=1, \dots, N} \cup \{(0, 0)\}$. Then, we say that I is an interpolation method WOWA-consistent if $I(w)$ is a monotonic function, and $I(w)(x) = x$ when $x = i/N$ for $i = 0, \dots, N$.

Definition 5 [7] Let \mathbf{p} and \mathbf{w} be two weighting vectors of dimension N , let I be a WOWA-consistent interpolation method; then, the mapping $WOWA: \mathbb{R}^N \rightarrow \mathbb{R}$ is a Weighted Ordered Weighted Averaging (WOWA) operator of dimension N if

$$WOWA_{\mathbf{p}, \mathbf{w}, I}(a_1, \dots, a_N) = WOWA_{\mathbf{p}, I(w)}(a_1, \dots, a_N),$$

2.1. Orness

Dujmovic [3] introduced in 1974 the degree of disjunction (or *orness*) for aggregation operators. Aggregation operators typically return a value between the minimum (used to model conjunction or *and*) and the maximum (used to model disjunction or *or*). Then, it is meaningful to consider in what extent one of such operators behave as the maximum (or the minimum). The orness is a measure of such extent (and the andness is a measure of

the extent in which they behave as the minimum). We review below the general definition of the *orness*. We also review a result about the orness of the OWA operator (the weight w as used above). We only consider the orness of w because the orness of the weighted mean (here represented by p) is always equal to 0.5.

Definition 6 Let \mathbb{C} be an aggregation operator in $[0,1]^N$ with parameter P ; then, the average value of \mathbb{C}_P is defined as

$$AV(\mathbb{C}_P) = \int_0^1 \dots \int_0^1 \mathbb{C}_P(a_1, \dots, a_N) da_1 \dots da_N.$$

Definition 7 [3] Let \mathbb{C} be an aggregation operator with parameters P ; then, the orness of \mathbb{C}_P is defined by

$$\text{orness}(\mathbb{C}_P) = \frac{AV(\mathbb{C}_P) - AV(\min)}{AV(\max) - AV(\min)}. \quad (1)$$

Proposition 8 [9] Let w be a weighting vector. Then, the orness of the OWA operator with respect to the weighting vector w is

$$\text{orness}(OWA_w) = \frac{1}{N-1} \sum_{i=1}^N (N-i)w_i.$$

2.2. Dispersion

One of the existing indices for weights is the degree of dispersion. In a few papers on weight determination, weights are often required to have a maximum dispersion (given a set of constraints, as e.g. an appropriate level of orness). Such maximum dispersion is required because it is considered inappropriate to accumulate the weights or importances into a single source (see discussion in Section 7.3 of [8]). Formally, the dispersion is defined in terms of the entropy.

Definition 9 Let $\mathbf{p} = (p_1, \dots, p_N)$ be a weighting vector; then, its entropy (dispersion) E is defined as (in this definition, $0 \ln 0$ is defined as zero):

$$E(\mathbf{p}) = \sum_{i=1}^N p_i \ln p_i.$$

For weighting vectors, the expression E is maximal when all the weights are equal to $p_i = 1/N$ and the maximal value obtained is $E(\mathbf{p}) = \log N$. In contrast, the minimal value $E(\mathbf{p}) = 0$ is obtained when $p_i = 1$ for one i .

A concept related to dispersion is variability. The variability is defined as the variance of the weights.

Definition 10 Let $\mathbf{p} = (p_1, \dots, p_N)$ be a weighting vector; then, its variance (variability) is defined as

$$\sigma^2(\mathbf{p}) = E[(\mathbf{p} - E[\mathbf{p}])^2].$$

So, the variability can be computed as follows:

$$\sigma^2(\mathbf{p}) = \frac{1}{N} \sum_{i=1}^N (p_i - E[\mathbf{p}])^2 = \frac{1}{N} \sum_{i=1}^N (p_i - \frac{1}{N})^2 = \sum_{i=1}^N \frac{p_i^2}{N} - \frac{1}{N^2}.$$

3. Sensitivity, orness, and dispersion

In a recent paper [7] we considered the problem of WOVA sensitivity with respect to different WOVA-consistent interpolation methods. We showed that it is possible to give bounds for the outcome of the WOVA operator (given weighting vectors p and q). Nevertheless, we also proved that with particular p and w , it is always possible to make the difference between the bounds as large as possible. Although this is the general case, we also showed that in general, the average difference between the outcomes of different WOVA operators with different interpolation methods is around 15% and in most of the cases lower.

In [7] we also studied the relationship between the average difference and the orness. Here we extend our previous analysis including the dispersion (or entropy) and the variance. The analysis needs a few additional definitions concerning the average difference. The definitions are listed below. In practice, we compute the average difference in a numerical way using a discretization of the interval.

Definition 11 [7] *Let w^1 and w^2 be two nondecreasing fuzzy quantifiers such that $w^1(x) \geq w^2(x)$ for all x in $[0, 1]$. Then, the average difference for a given weighting vector p is defined as:*

$$\Delta WOVA_{p, w^1, w^2} = \int_{x \in [0, 1]^N} (WOVA_{\mathbf{p}, w^1}(x) - WOVA_{\mathbf{p}, w^2}(x)) dx$$

In addition to this definition, we have also considered the average difference for all weighting vectors p in $[0, 1]^N$. The definition is given below.

Definition 12 [7] *Let w^1 and w^2 be two nondecreasing fuzzy quantifiers such that $w^1(x) \geq w^2(x)$ for all x in $[0, 1]$. Then, the average difference of WOVA for all p is defined as:*

$$\Delta^p WOVA_{w^1, w^2} = \int_{p \in WV^N} \int_{x \in [0, 1]^N} (WOVA_{\mathbf{p}, w^1}(x) - WOVA_{\mathbf{p}, w^2}(x)) dx dp$$

where WV defines the set of weighting vectors of dimension N .

In the case that it is not possible to ensure $w^1(x) \geq w^2(x)$, we will use the following alternative definition [7],

$$\Delta WOVA'_{p, w^1, w^2} = \sqrt{\int_{x \in [0, 1]^N} (WOVA_{\mathbf{p}, w^1}(x) - WOVA_{\mathbf{p}, w^2}(x))^2 dx},$$

with the following average difference [7]:

$$\overline{\Delta^p W O W A'}_{w^1, w^2} = \int_{p \in W^{V^N}} \overline{\Delta W O W A'}_{p, w^1, w^2} dp$$

In [7], we considered the maximum average difference (for a set of weighting vectors p), as well as the average difference for a set of different quantifiers. We reproduce in Table 1 such results for the case of $N = 3$ and for the quantifiers w_w^l , w_w^h , w^* , and w_w^+ . They are, respectively, the lowest function constructed from w , the highest function constructed from w , the interpolated function from w as in [6], and the linear interpolation [1,2]. These data have been computed numerically. No numerical expression has been found for the average differences of the WOVA operator.

In addition, Table 1 gives the three indices (orness, entropy, and variance) for a set of weighting vectors of dimension 3. The table also includes the correlation coefficient of the variables with respect to the three indices. It can be seen that the entropy is not correlated with the other indices, but that it is with respect to the average difference (except for the last column).

The results show that the maximum average difference corresponds to the pairs (w_w^h, w_w^l) , which are the most extreme quantifiers. Also, that the average difference between w^* and the linear interpolation is lower than the difference between w^* and the extreme quantifier. We can also see that the orness explains most of the average differences for the first three cases considered, and that neither the entropy nor the variance do.

In the case of the difference between the linear interpolation and the w^* , the following linear model results (we have computed it using the R project),

$$\Delta^p W O W A_{w_w^*, w_w^+} = -0.030887 + 0.057062 \text{orness},$$

with residuals in the interval $[-0.0144590, 0.0140718]$ and a p-value lower than $2.2 \cdot 10^{-16}$. So, it is clear that the orness has a large influence on the average difference. Adding entropy as an additional variable does not change much these results. The corresponding model is:

$$\Delta^p W O W A_{w_w^*, w_w^+} = -0.0306076 + 0.0570616 \text{orness} - 0.0003997 \text{entropy}$$

In the case of $\Delta^p W O W A'_{w_w^*, w_w^+}$, the variance (and in a less extent, entropy) is the main variable in the model and the orness has a less relevant role:

$$\Delta^p W O W A'_{w_w^*, w_w^+} = 0.025585 - 0.001899 \text{orness} + 0.147299 \text{variance} - 0.012247 \text{entropy}$$

These results into a p-value of $3.1 \cdot 10^{-5}$, and residuals in the interval $[-0.037006, 0.024344]$. Similar residuals and p-values are obtained in the case of a model with variance and entropy, or in the case of a model with variance alone.

| w | orness | entropy | variance | $\Delta PWOVA_{w_m^h, w_m^l}$ | $\Delta PWOVA_{w_m^h, w_m^*}$ | $\Delta PWOVA_{w_m^*, w_m^+}$ | $\Delta PWOVA_{w_m^*, w_m^+}$ |
|--------------------|--------|---------|----------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|
| (0,0,0,1,0) | 0.000 | 0.000 | 0.222 | 0.144 | 0.125 | -0.042 | 0.072 |
| (0,0,0,125,0.875) | 0.063 | 0.377 | 0.149 | 0.143 | 0.110 | -0.029 | 0.047 |
| (0,0,0,25,0.75) | 0.125 | 0.562 | 0.097 | 0.142 | 0.098 | -0.019 | 0.029 |
| (0,0,0,375,0.625) | 0.188 | 0.662 | 0.066 | 0.142 | 0.088 | -0.011 | 0.017 |
| (0,0,0,5,0.5) | 0.250 | 0.693 | 0.056 | 0.141 | 0.078 | -0.003 | 0.006 |
| (0,0,0,625,0.375) | 0.313 | 0.662 | 0.066 | 0.140 | 0.081 | -0.007 | 0.018 |
| (0,0,0,75,0.25) | 0.375 | 0.562 | 0.097 | 0.139 | 0.077 | -0.005 | 0.019 |
| (0,0,0,875,0.125) | 0.438 | 0.377 | 0.149 | 0.138 | 0.074 | -0.004 | 0.020 |
| (0,0,1,0,0,0) | 0.500 | 0.000 | 0.222 | 0.138 | 0.071 | -0.003 | 0.020 |
| (0,125,0,0,0.875) | 0.125 | 0.377 | 0.149 | 0.143 | 0.112 | -0.032 | 0.063 |
| (0,125,0,125,0.75) | 0.188 | 0.736 | 0.087 | 0.143 | 0.099 | -0.022 | 0.037 |
| (0,125,0,25,0.625) | 0.250 | 0.900 | 0.045 | 0.142 | 0.090 | -0.014 | 0.021 |
| (0,125,0,375,0.5) | 0.313 | 0.974 | 0.024 | 0.141 | 0.082 | -0.008 | 0.012 |
| (0,125,0,5,0.375) | 0.375 | 0.974 | 0.024 | 0.140 | 0.078 | -0.006 | 0.013 |
| (0,125,0,625,0.25) | 0.438 | 0.900 | 0.045 | 0.139 | 0.075 | -0.004 | 0.014 |
| (0,125,0,75,0.125) | 0.500 | 0.736 | 0.087 | 0.139 | 0.071 | -0.002 | 0.016 |
| (0,125,0,875,0.0) | 0.563 | 0.377 | 0.149 | 0.138 | 0.069 | -0.001 | 0.020 |
| (0,25,0,0,0.75) | 0.250 | 0.562 | 0.097 | 0.143 | 0.098 | -0.022 | 0.056 |
| (0,25,0,125,0.625) | 0.313 | 0.900 | 0.045 | 0.142 | 0.089 | -0.015 | 0.030 |
| (0,25,0,25,0.5) | 0.375 | 1.040 | 0.014 | 0.141 | 0.080 | -0.007 | 0.014 |
| (0,25,0,375,0.375) | 0.438 | 1.082 | 0.003 | 0.140 | 0.075 | -0.004 | 0.007 |
| (0,25,0,5,0.25) | 0.500 | 1.040 | 0.014 | 0.140 | 0.072 | -0.003 | 0.011 |
| (0,25,0,625,0.125) | 0.563 | 0.900 | 0.045 | 0.139 | 0.069 | -0.001 | 0.014 |
| (0,25,0,75,0.0) | 0.625 | 0.562 | 0.097 | 0.138 | 0.066 | 0.000 | 0.019 |
| (0,375,0,0,0.625) | 0.375 | 0.662 | 0.066 | 0.142 | 0.085 | -0.012 | 0.051 |
| (0,375,0,125,0.5) | 0.438 | 0.974 | 0.024 | 0.141 | 0.078 | -0.007 | 0.024 |
| (0,375,0,25,0.375) | 0.500 | 1.082 | 0.003 | 0.141 | 0.072 | -0.003 | 0.010 |
| (0,375,0,375,0.25) | 0.563 | 1.082 | 0.003 | 0.140 | 0.069 | -0.001 | 0.007 |
| (0,375,0,5,0.125) | 0.625 | 0.974 | 0.024 | 0.139 | 0.065 | 0.001 | 0.013 |
| (0,375,0,625,0.0) | 0.688 | 0.662 | 0.066 | 0.138 | 0.063 | 0.002 | 0.018 |
| (0,5,0,0,0.5) | 0.500 | 0.693 | 0.056 | 0.141 | 0.071 | -0.002 | 0.049 |
| (0,5,0,125,0.375) | 0.563 | 0.974 | 0.024 | 0.141 | 0.066 | 0.002 | 0.024 |
| (0,5,0,25,0.25) | 0.625 | 1.040 | 0.014 | 0.140 | 0.064 | 0.002 | 0.013 |
| (0,5,0,375,0.125) | 0.688 | 0.974 | 0.024 | 0.139 | 0.062 | 0.003 | 0.012 |
| (0,5,0,5,0.0) | 0.750 | 0.693 | 0.056 | 0.138 | 0.065 | -0.003 | 0.006 |
| (0,625,0,0,0.375) | 0.625 | 0.662 | 0.066 | 0.141 | 0.058 | 0.008 | 0.050 |
| (0,625,0,125,0.25) | 0.688 | 0.900 | 0.045 | 0.140 | 0.054 | 0.010 | 0.029 |
| (0,625,0,25,0.125) | 0.750 | 0.900 | 0.045 | 0.139 | 0.054 | 0.009 | 0.021 |
| (0,625,0,375,0.0) | 0.813 | 0.662 | 0.066 | 0.139 | 0.055 | 0.006 | 0.017 |
| (0,75,0,0,0.25) | 0.750 | 0.562 | 0.097 | 0.140 | 0.045 | 0.018 | 0.054 |
| (0,75,0,125,0.125) | 0.813 | 0.736 | 0.087 | 0.139 | 0.044 | 0.017 | 0.035 |
| (0,75,0,25,0.0) | 0.875 | 0.562 | 0.097 | 0.139 | 0.045 | 0.014 | 0.028 |
| (0,875,0,0,0.125) | 0.875 | 0.377 | 0.149 | 0.140 | 0.031 | 0.029 | 0.061 |
| (0,875,0,125,0.0) | 0.938 | 0.377 | 0.149 | 0.139 | 0.033 | 0.025 | 0.045 |
| (1,0,0,0,0.0) | 1.000 | 0.000 | 0.222 | 0.139 | 0.018 | 0.039 | 0.069 |
| Average difference | | | | 0.1402 | 0.0716 | -0.0024 | 0.0273 |
| Corr w/ orness | 1.000 | 0.000 | 0.000 | -0.753 | -0.961 | 0.922 | -0.025 |
| Corr w/ entropy | 0.000 | 1.000 | -0.981 | 0.000 | 0.006 | -0.008 | -0.651 |
| Corr w/ variance | 0.000 | -0.981 | 1.000 | 0.000 | -0.005 | 0.007 | 0.656 |

Table 1. Orness, entropy, variance, and average differences (for all weights p) for a few weighting vectors of dimension 3, and their correlation coefficients with differences.

4. Conclusions

In this paper we have further studied the sensitivity of the WOVA operator when we change the interpolation method used when constructing the function w^* . We have seen that the average difference is lower than 10% in general, and with an average of 2% when comparing the original definition and a linear interpolation. Maximum average difference is obtained for the extreme quantifiers w_w^h and w_w^l .

In addition, we have compared the average difference with the orness, the entropy and the variance. We have seen that the orness explains most of the average difference for $\overline{\Delta^p W O W A}$. In contrast, for $\overline{\Delta^p W O W A'}_{w_w^*, w_w^+}$ the variance and the entropy explain the main difference but in a lesser extend.

We have also computed linear models that would permit a user to estimate the sensitivity of the WOWA for a given weighting vector w . These results would enables us to develop methods for parameter determination that take into account the sensitivity of the operator. For example, it is possible to find the parameters for a given orness that have the lowest sensitivity. With such constraints, the weighting vector w would be the one with maximum entropy (or dispersion).

Although the results have been presented for weighting vectors of dimension $N = 3$, similar results apply to vectors of dimension $N = 2$. For these two dimensions, exhaustive computations have been done for all weights p and w listed in Table 1. In the experiments, 100 different values have been considered in the input domain (the unit interval).

As future work we will consider the effect of other indices in the explanation of the average difference between WOWA outcomes.

Acknowledgments

Partial support by the Generalitat de Catalunya (2005 SGR 00446 and 2005-SGR-00093) and by the Spanish MEC (projects ARES – CONSOLIDER INGENIO 2010 CSD2007-00004 – and eAEGIS – TSI2007-65406-C03-02) is acknowledged. The author thanks Zhenbang Lv for his comments.

References

- [1] Ceravolo, P., Damiani, E., Viviani, M. (2006) Adding a trust layer to semantic web metadata, in E. Herrera-Viedma, G. Pasi, F. Crestani (eds) *Soft Computing in Web Information Retrieval: Models And Applications*, Springer, 87-104.
- [2] Damiani, E., di Vimercati, S. D. C., Samarati, P., Viviani, M. (2006) A WOWA-based aggregation technique on trust values connected to metadata, *Electronic Notes in Theoretical Computer Science* 157 131-142.
- [3] Dujmović, J. J. (1974) Weighted conjunctive and disjunctive means and their application in system evaluation, *Journal of the University of Belgrade, EE Dept., Series Mathematics and Physics* 483 147-158.
- [4] Torra, V. (1996) Weighted OWA operators for synthesis of information, *Proc. of the 5th IEEE Int. Conf. on Fuzzy Systems* 966-971.
- [5] Torra, V. (1997) The weighted OWA operator, *Int. J. of Intel. Syst.* 12 153-166.
- [6] Torra, V. (2000) The WOWA operator and the interpolation function W^* : Chen and Otto's interpolation method revisited, *Fuzzy Sets and Systems* 113:3 389-396.
- [7] Torra, V., Lv, Z. (2008) On the WOWA operator and its interpolation function, submitted.
- [8] Torra, V., Narukawa, Y. (2007) *Modeling decisions: information fusion and aggregation operators*, Springer.
- [9] Yager, R. R. (1988) On ordered weighted averaging aggregation operators in multi-criteria decision making, *IEEE Trans. on Systems, Man and Cybernetics* 18 183-190.
- [10] Yager, R. R. (1996) Quantifier guided aggregation using OWA operators, *Int. J. of Intel. Syst.* 11 49-73.

This page intentionally left blank

Keyword Index

| | | | |
|--|-----------------------|---|----------|
| active contours | 355 | discretization | 161 |
| affinity | 283 | dispersion | 430 |
| aggregation function | 273 | dissolution | 51 |
| aggregation operator(s) | 206, 430 | distributed constraints | 118 |
| ant colony systems | 333 | electronic market | 33 |
| arbitrage | 33 | ensemble methods | 178 |
| artificial intelligence in medical applications | 178 | entropy | 118, 215 |
| artificial intelligence in medicine | 323 | ergonomics | 186 |
| auctions | 70 | evolution strategies | 253 |
| autonomous mobile robots | 380 | evolutionary algorithms | 253, 342 |
| bartering | 24, 33 | exogenous clustering based on rules | 170 |
| basal cell carcinoma | 178 | expectation maximization | 363 |
| case-based reasoning | 178, 186, 323, 372 | experimentation | 41 |
| CBR | 151 | explanations | 151 |
| classification | 244 | fault identification | 421 |
| clinical test | 170 | fault isolation | 138, 421 |
| cluster ensemble | 161 | fault location | 225 |
| clustering | 151, 323 | fuzzy description logics | 411 |
| CN2-SD | 225 | fuzzy sets theory | 333 |
| coalitions | 51 | genetic algorithms | 253 |
| combinatorial auctions | 303 | graded BDI agent model | 41 |
| complete matching | 138 | grasp | 128 |
| computational ecologies | 293 | heuristic function | 342 |
| computer aided systems | 323 | idempotent | 206 |
| computer games | 342 | image | 197 |
| computer vision | 398 | information | 215 |
| consistency techniques | 421 | intelligent monitoring | 236 |
| conversational agents | 60 | interval analysis | 262 |
| data complexity | 244 | irreducibility | 273 |
| data mining | 236, 253 | Kalman filter | 363 |
| decision support and knowledge management | 170 | kernel methods | 262 |
| depth from zoom | 355 | knowledge-based applications in medicine | 170 |
| dermatoscopy | 178 | knowledge discovery | 151, 170 |
| description logics | 411 | Kohonen self-organizing map | 313 |
| digital business ecosystems | 293 | learning classifier systems | 253 |
| dimensionality | 244 | learning policies | 342 |
| directory services | 24 | localization | 388 |
| discrete copula | 273 | machine learning | 225 |
| discrete quasi-copula | 273 | magnitude order spaces | 215 |
| | | matrix representation | 273 |

| | | | |
|------------------------------------|--------------------------------|---|-----------|
| melanoma(s) | 178, 323 | reflectance confocal microscopy | 178 |
| mobile robots | 388 | regular sequences | 128 |
| model-based | 421 | rehabilitation | 170 |
| model-based fault diagnosis | 138 | reputation | 197 |
| motion analysis | 398 | resource allocation | 70 |
| motion estimation | 355 | response time variability | 128 |
| motion segmentation | 398 | robustness | 70 |
| moulding process | 236 | ruled quadric surface | 206 |
| multiagent platforms | 7 | satisfiability | 109 |
| multi-agent systems | 7, 16, 51, 60, 88, 197, 372 | scheduling | 128 |
| multi-layer perceptron | 313 | scrabble | 342 |
| multiple-valued logic | 60 | short-term electric consumption | 313 |
| multi-start metaheuristic | 128 | simulation | 88 |
| mutual information | 161 | SLAM | 363 |
| negotiating agents | 293 | social networks | 283 |
| negotiation | 51 | social norms | 88 |
| negotiation styles | 293 | spatial reasoning | 380 |
| normative implementation | 16 | structural analysis | 138 |
| normative language | 16 | subgroup discovery | 225 |
| normative systems | 16 | supervised learning | 253 |
| object recognition | 388 | symmetric | 206 |
| one red paperclip | 33 | synthetic data sets | 244 |
| orness | 430 | the nearest neighbor technique | 313 |
| partial deduction | 60 | t-norm based fuzzy logics without and with truth constants | 411 41 |
| particle filter | 363 | tourism | 41 |
| pattern discovery | 323 | traumatic brain injury | 170 |
| planning | 342 | traveling salesman | 333 |
| pooled voting | 161 | trust | 51 |
| power quality monitoring | 225 | TSP | 333 |
| principal component analysis | 236 | uncertainty | 421 |
| privacy | 118 | virtual organizations | 16 |
| probabilistic dynamic belief logic | 197 | virtual power plant | 313 |
| qualitative reasoning | 215, 262, 380 | voltage sag (dip) source location | 225 |
| qualitative trajectories | 380 | web crawling | 283 |
| random SAT models | 109 | web mining | 283 |
| real-time | 372 | winner determination problem | 303 |
| recommender system(s) | 41, 293 | WOWA | 430 |
| redundancy | 138 | | |

Author Index

| | | | |
|---------------------------|----------|--------------------------|----------|
| Acíar, S. | 51 | Fornells, A. | 178, 323 |
| Agell, N. | 215, 262 | García Gorrostieta, J.M. | 186 |
| Aguiló, I. | 273 | García-Cerdaña, À. | 411 |
| Alberola, J.M. | 7 | García-Fornes, A. | 7 |
| Alenyà, G. | 355 | García-Molina, A. | 170 |
| Alquezar, R. | 342 | García-Rudolph, A. | 170 |
| Alsinet, T. | ix | García-Villoria, A. | 128 |
| Angulo, C. | 262 | Garriga, C. | 323 |
| Ansótegui, C. | 99, 109 | Geffner, H. | 4 |
| Argente, E. | 16 | Gelso, E.R. | 138, 421 |
| Armengol, E. | 151 | Gibert, K. | 161, 170 |
| Armengol, J. | 138, 421 | Godo, L. | 41 |
| Aulinas, J. | 363 | Golobardes, E. | 178, 323 |
| Baeza-Yates, R. | 3 | Gómez-Sebastià, I. | 161 |
| Barrera, V. | 225 | González López, S. | 186 |
| Becceneri, J.C. | 333 | Gonzalez-Romero, A. | 342 |
| Béjar, R. | 99 | Grieu, S. | 313 |
| Berjaga, X. | 236 | Heras, S. | 372 |
| Bernabeu, M. | 170 | Hormazábal, N. | 51, 293 |
| Bernadó-Mansilla, E. | 244, 253 | Julián, V. | 16, 372 |
| Bonet, M.L. | 109 | Levy, J. | 109 |
| Botti, V. | 7, 16 | Lladó, X. | 363, 398 |
| Brito, I. | 118 | Lopardo, G. | 293 |
| Busquets, D. | 70 | López, B. | 225 |
| Cabanillas, D. | 24, 33 | Lopez de Mantaras, R. | 388 |
| Campos, J. | 80 | Lopez-Sanchez, M. | 80 |
| Canaleta, X. | 283 | Macià, N. | 244 |
| Carrera, C. | 178 | Malveyh, J. | 178, 323 |
| Casali, A. | 41 | Mateu, C. | 99 |
| Castillo, S.M. | 138, 421 | Meléndez, J. | 225, 236 |
| Chieppa, A. | 161 | Meseguer, P. | 118 |
| Corominas, A. | 128 | Montaner, M. | 293 |
| Cortés, U. | 24, 33 | Morales-Ortigosa, S. | 253 |
| Criado, N. | 16 | Morveli-Espinoza, M. | 60 |
| da Luz, E.F.P. | 333 | Muñoz, V. | 70, 303 |
| de la Rosa, J.L. | 293 | Murillo, J. | 303 |
| de la Rosa i Esteve, J.L. | 51 | Musavi, A. | 80 |
| Dellunde, P. | 197 | Navarro, M. | 372 |
| Escrig, M.T. | 380 | Nicolas, R. | 178, 323 |
| Espinosa, A. | 7 | Orduña Cabrera, F. | 186 |
| Esteva, F. | 411 | Orriols-Puig, A. | 244, 253 |
| Esteva, M. | 80 | Pallares, A. | 236 |
| Fernández, C. | 99 | Palou, J. | 178 |

| | | | |
|------------------|----------|-------------------|----------|
| Pastor, R. | 128 | Sánchez, M. | 215 |
| Peris, J.C. | 380 | Sánchez-Marrè, M. | 161, 186 |
| Petillot, Y. | 363 | Sandri, S. | 333 |
| Pinyol, I. | 197 | Segura, S. | 178 |
| Plana, J. | 380 | Sierra, C. | 41 |
| Polit, M. | 313 | Such, J.M. | 7 |
| Prats, F. | 215 | Suñer, J. | 273 |
| Puig, S. | 178, 323 | Toledo, R. | 388 |
| Puyol-Gruart, J. | ix, 60 | Tormos, J.M. | 170 |
| Ramisa, A. | 388 | Torra, V. | 430 |
| Recasens, J. | 206 | Torras, C. | ix, 355 |
| Ribes, A. | 388 | Torrens, J. | 273 |
| Roig-Rovira, T. | 170 | Tran, V.G. | 313 |
| Ros, P. | 283 | Vallejo, A. | 283 |
| Roselló, L. | 215 | Vernet, D. | 283, 323 |
| Ruiz, F.J. | 262 | Villatoro, D. | 88 |
| Sabater-Mir, J. | 88, 197 | Willmott, S. | 24, 33 |
| Salvi, J. | 363, 398 | Zaballos, A. | 283 |
| Sánchez, J. | 225 | Zappella, L. | 398 |